

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА ДИЗАЙНУ

ФАКУЛЬТЕТ МЕХАТРОНИКИ ТА КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розроблення математичного та програмного забезпечення для
автоматизованого проєктування декоративних елементів для виробів
шкіргалантереї

Рівень вищої освіти	<u>другий (магістерський)</u>
Спеціальність 122	<u>Комп'ютерні науки</u>
Освітня програма	<u>Комп'ютерні науки</u>

Виконав: студент групи МГЗІТ-22

Ярослав КОНЕЦЬКИЙ

Науковий керівник: к.т.н. Наталія ЧУПРИНКА
Рецензент д.т.н., проф. Сергій КРАСНИТСЬКИЙ

Київ 2023

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ТЕХНОЛОГІЙ ТА
ДИЗАЙНУ

Факультет мехатроніки та комп'ютерних технологій
Кафедра комп'ютерні науки
Рівень вищої освіти другий (магістерський)
Спеціальність 122 Комп'ютерні науки
Освітня програма Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

_____ Володимир ЩЕРБАНЬ

«___» _____ 2023__ року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТА

Конецькому Ярославу Олегу Миколайовичу

1. Тема роботи: Розроблення математичного та програмного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї

Науковий керівник роботи: Чупринка Наталія Вікторівна, к.т.н.
затверджені наказом закладу вищої освіти від 12.09.2023 року, № 210-уч.

2. Вихідні дані до кваліфікаційної роботи:

Розробка кафедри комп'ютерних наук

3. Зміст кваліфікаційної роботи (перелік питань, які потрібно розробити)
Вступ; РОЗДІЛ 1. Шкіргалантерейні вироби; РОЗДІЛ 2. Розроблення математичного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї; РОЗДІЛ 3. Розроблення програмного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї. Додатки – програмні коди модулів системи.

4. Дата видачі завдання 08.2023р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної магістерської роботи	Терміни виконання етапів	Примітка про виконання
1	Вступ	30.08.2023	
2	Розділ 1 Шкіргалантерейні вироби	06.09.2023	
3	Розділ 2. Розроблення математичного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї	28.09.2023	
4	Розділ 3. Розроблення програмного забезпечення для автоматизованого проектування декоративних елементів для декоративних елементів для	21.10.2023	
5	Висновки	29.10.2023	
6	Оформлення кваліфікаційної роботи (чистовий варіант)	06.11.2023	
7	Подача кваліфікаційної роботи (проєкту) науковому керівнику для відгуку (за 14 днів дозахисту)		
8	Подача кваліфікаційної роботи (проєкту) для рецензування		
9	Перевірка кваліфікаційної роботи на наявність ознак плагіату		
10	Подання кваліфікаційної роботи на затвердження завідувачу кафедри		

З завданням ознайомлений:

Студент _____

Ярослав КОНЕЦЬКИЙ

Науковий керівник _____

Наталія ЧУПРИНКА

АНОТАЦІЯ

Конецький Я. М. Розроблення математичного та програмного забезпечення для автоматизованого проєктування декоративних елементів для виробів шкіргалантереї – Рукопис.

Дипломна магістерська робота за спеціальністю 122- «Комп'ютерні науки» – Київський національний університет технологій та дизайну, Київ, 2023 рік.

В роботі запропонований методи та алгоритми для проєктування декоративних елементів для виробів шкіргалантереї. Запропоновані алгоритми реалізовані в програмний продукт для автоматизованого проєктування декоративних елементів для виробів шкіргалантереї.

Розроблений програмний продукт має дружній інтерфейс та не потребує спеціальних знань з комп'ютерної техніки для роботи з ним.

Ключові слова: математичне та програмне забезпечення, декоративні елементи, шкіргалантерейні вироби, автоматизоване проєктування

ABSTRACT

Konetskyi Y. M. Development of mathematical and software for automated design of decorative elements for leather goods - Manuscript.

Master's thesis in specialty 122- "Computer Science" - Kyiv National University of Technology and Design, Kyiv, 2023. The work proposes methods and algorithms for designing decorative elements for leather goods.

The proposed algorithms are implemented in a software product for the automated design of decorative elements for leather goods. The developed software product has a friendly interface and does not require special knowledge of computer technology to work with it.

Keywords: mathematical and software, decorative elements, leather goods, automated design

Зміст

Вступ.....	6
1. ШКІРГАЛАНТЕРЕЙНІ ВИРОБИ.....	8
2. РОЗРОБЛЕННЯ МАТЕМАТИЧНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ДЕКОРАТИВНИХ ЕЛЕМЕНТІВ ДЛЯ ВИРОБІВ ШКІРГАЛАНТЕРЕЙ.....	18
3. РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ДЕКОРАТИВНИХ ЕЛЕМЕНТІВ ДЛЯ ВИРОБІВ ШКІРГАЛАНТЕРЕЙ	32
Висновки.....	49
Список використаних джерел.....	50
Додатки.....	54

Вступ

Найважливіші напрями розвитку шкіряної галузі бачаться у зміні асортиментної політики вітчизняних підприємств, у кооперації з взуттєвими фірмами, і в той же час у скороченні частки виробництва шкір для верху взуття

Щоб вийти на передові позиції в галузі, використовуючи напрацьований практикою арсенал прийомів конкурентної боротьби, шкіряне підприємство зобов'язане насамперед оцінити свої рейтинги можливості позиціонування на ринку. Завдання було б значною мірою спрощено, якби вітчизняні шкіряні підприємства зуміли знайти форми організаційно-економічного об'єднання, щоб залишити кризовий стан «одинаків». у тому числі шкіряно-взуттєвий.

Для виготовлення галантерейних товарів застосовують найрізноманітнішу сировину - текстильні матеріали, натуральну та штучну шкіру, пластичні маси, дерево, скло, тому прогрес у виробництві групи, що розглядається, пов'язаний як з розширенням сировинної бази, так і з удосконаленням технологічних процесів.

Самооцінка підприємства, що проводиться керівниками, економістами, маркетингологами та іншими провідними фахівцями, покликана не так консервувати його становище в галузі, як виявити перспективи розвитку, навіть в умовах несприятливого зовнішнього середовища. При цьому недоліки та прорахунки у внутрішньому середовищі мають своєчасно виявляти та долати. У керівництва підприємства завжди є можливість їх уникати, і вплив довкілля у разі немає вирішального значення.

Безпосередньо на конкурентоспроможність вітчизняних шкіргалантерейних виробів впливають такі чинники як: структура та властивості матеріалів; конструкція моделей виробів; якість виготовлення. Вітчизняна продукція загалом не поступається імпортною.

Це свідчить про досить прийнятний рівень професіоналізму вітчизняних фахівців – модельєрів.

Щоб досягти успіху в умовах жорсткої конкуренції на ринку, вітчизняним підприємствам належить не просто вдосконалити асортимент, моделювання і дизайн продукції, що випускається, а й вирішити глибше завдання - впровадження інформаційних технологій у виробництво.

Метою даної дипломної є розробка математичного та програмного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї.

Об'єкт дослідження – процес проектування декоративних елементів для шкіргалантереї.

Методи дослідження. Теоретичні дослідження роботи ґрунтуються на основних положеннях проектування деталей виробів шкіргалантереї та застосування комп'ютерних наук при вирішенні поставленої задачі.

Експериментальні дослідження заключалися в тестуванні розробленого продукту для проектування автоматизованого проектування декоративних елементів для виробів шкіргалантереї.

Наукова новизна полягає у розробці математичного та програмного забезпечення для автоматизованого проектування виробів шкіргалантереї.

Апробація результатів магістерської роботи. Основні положення і результати магістерської роботи протягом 2023 року були представлені та одержали позитивну оцінку на міжнародній науковій конференції.

Публікації. За темою магістерської роботи «Розроблення математичного та програмного забезпечення для автоматизованого проектування декоративних елементів для виробів шкіргалантереї» опубліковано одна наукових робота.

1. ШКІРГАЛАНТЕРЕЙНІ ВИРОБИ

1.1. Класифікація шкіргалантерейних виробів

Сумки – невід'ємні деталі одягу. Як багато іншого в побуті людини, ці чисто утилітарні вироби з часом набули також функції прикраси та доповнення костюма. Вони, як взуття, головні убори та ювелірні вироби, є підлеглими деталями, вибір яких залежить насамперед від костюма.

Шкіргалантерейні вироби в залежності від умов використання поділяють на побутові та спеціальні.

Побутові вироби - це шкіргалантерейні товари, що служать людині в побуті та виконують утилітарні та естетичні функції.

Спеціальні вироби – це вироблені шкіргалантереї, що використовуються в спеціальних умовах, що мають особливі конструкції та службовці людині для виконання будь-якої роботи, операцій.

Як побутові, так і спеціальні шкіргалантерейні вироби за призначенням поділяють на дві групи:

- для перенесення предметів;
- для зберігання предметів.

За статево-віковою ознакою споживачів шкіргалантерейні вироби поділяються на жіночі, чоловічі, молодіжні та дитячі.

Найбільшою групою є вироби для перенесення та зберігання предметів, тобто. сумки.

За умовами експлуатації сумки поділяють на повсякденні, вихідні, господарські, дорожні, спортивні.

За видом оформлення сумки поділяються на ошатні, ділові, офіційні.

За сезонами сумки ділять на весняно-літні, осінньо-зимові.

1.2 Асортимент шкіргалантерейних виробів

Найбільшу групу шкіргалантерейних виробів представляють жіночі сумки. Жіночі ошатні сумки (театральні) відрізняються витонченим оформленням та оздобленням. Вони мають невеликі розміри, різноманітні прикраси та декоративні деталі – аплікацію, плетіння, вишивку, оригінальну фурнітуру. Вони можуть бути предметом декоративно-прикладного мистецтва, тобто. мати художню цінність та водночас практичне застосування.

Жіночі повсякденні сумки призначені задовольняти різноманітні вимоги залежно від характеру вкладених предметів (ділові, загального призначення), умов їх використання (весняно-літні, осінньо-зимові), стилю (класичні, спортивні, романтичні, фольклорні) та ін. кількість видів та конструкцій даних виробів. Оскільки група жіночих сумок велика, із неї часто виділяють молодіжні сумки, особливістю яких є авангардний стиль. У зв'язку з швидкою змінністю моди конструкція та оформлення молодіжних сумок найбільш схильні до змін.

Сумки для косметики характеризуються малими розмірами та простою конструкцією.

Дорожні сумки різноманітніші за конструкціями та видами, ніж господарські. Для них характерні об'ємні зовнішні та внутрішні кишені, стяжні ремені, особливо міцне кріплення деталей та фурнітури. Різновидами дорожніх сумок є портплед та саквож. Популярні конструкції дорожніх сумок з об'ємом, що трансформується, на коліщатках.

Спортивні сумки близькі за конструкцією та розмірами до господарських та дорожніх. Їх, як правило, відрізняє характерне оформлення з використанням шовкографії, різних емблем, фурнітури тощо, а також яскрава кольорова гама та наявність специфічних вузлів та деталей, що підкреслюють спортивну приналежність виробу.

Сумки господарські мають спрощену конструкцію, їх найчастіше випускають із синтетичних матеріалів з підкладкою, що миється. Літні сумки

відрізняються яскравою кольоровою гамою, простою конструкцією, використанням нетрадиційних матеріалів (тканин, плетінь із соломки тощо).

Сумки для учнів – асортиментна група, яка в останні роки замінила учнівські портфелі. Вони мають конструкції, зручні для розміщення підручників, зошитів, шкільно-письмового приладдя.

Дитячі сумки поділяють на два види: для дітей шкільного та дошкільного віку. Вони відрізняються розмірами, оформленням та конструкцією, що визначає призначення виробів. Оформлення сумок для дошкільнят часто має ігровий характер. Чоловічі сумки з'явилися порівняно недавно.

Візитки – невеликі сумки для документів, обмежена кількість ділових паперів, особистих речей.

Чоловічі сумки стали досить великою асортиментною групою загального призначення, що включає практичні об'ємні вироби звичайного строгого оформлення.

Ділові папки призначені для ділових паперів, книг, журналів. Бюварні папки розміщуються на письмовому столі та служать для зберігання поштових паперів, конвертів, різної кореспонденції.

Адресні папки використовуються для вкладання вітальної ювілейної адреси та характеризуються відповідним оформленням. Папки мають прямокутний силует та розміри, що визначаються розмірами ділових паперів.

Бюварні та адресні папки – плоскі вироби, а ділові папки можуть мати певний обсяг. Великою популярністю у ділових людей користуються органайзери – папки невеликого розміру з блокнотом-тижневиком, телефонною книжкою, існують відділення для візитних карток, ручок та олівців, ділових паперів, грошей тощо.

Дорожні вироби в асортименті шкіргалантерейних товарів переважно представлені валізами. Валізи призначені для перевезення та зберігання

особистих речей та предметів побуту. Розрізняють дорожні валізи та валізи-дипломати.

Значне місце в асортименті шкіргалантерейних виробів займають портфелі. Залежно від призначення портфелі ділять на ділові, дорожні та учнівські, а також на чоловічі та жіночі.

Ділові портфелі призначені для повсякденного перенесення ділових паперів, журналів, канцприладдя. Їх характерний невеликий обсяг, відповідні розміри, наявність відділень для ділових паперів, книжок тощо., строгі лінії деталей і силуету, стримана кольорова гама.

Жіночі ділові портфелі відрізняються меншими розмірами та об'ємом, більш різноманітною кольоровою гамою матеріалів, що використовуються, і складнішим кроєм зовнішніх додаткових деталей.

Дорожні портфелі призначені для перевезення невеликої кількості книг, ділових паперів, документів та частково особистих речей. Такий виріб характеризується великим обсягом, розмірами порівняно з діловим портфелем, наявністю відділень для ділових паперів, застосуванням об'ємних зовнішніх кишень, додаткових ременів, гортів, зтяжників, стриманою кольоровою гамою матеріалів.

Учнівські портфелі призначені для носіння підручників, шкільно-письмового приладдя. Розрізняють учнівські портфелі 1, 2 та 3 груп для учнів відповідно 1-3, 4-6, 7-11 класів.

Учнівський портфель характеризується обсягом та розмірами, відповідними розмірам шкільно-письмового приладдя та підручників, які в ньому носять, простою конструкцією, наявністю 2-3 відділень або зовнішньої кишені, яскравою кольоровою гамою матеріалів (1 група), строгим силуетом корпусу та лініями деталей, наявністю знімних ременів для носіння портфелів за спиною учнями початкових класів.

Шкільні ранці служать для носіння підручників та шкільно-письмового приладдя учнями 1-3 класів. Ці вироби характеризуються певним обсягом та розмірами, простою конструкцією, закриванням на клапан або застібку блискавку, яскравим кольоровим оформленням. Останнім часом ранці майже повністю витіснені рюкзаками.

Дрібні шкіргалантерейні вироби включають такі види:

- портмоне – для зберігання паперових грошей та розмінної монети;
- гаманці – для зберігання документів, дрібних паперів та паперових грошей;
- гаманці – для розмінної монети;
- футляри – для зберігання ключів, окулярів тощо;
- обкладинки – для запобігання пошкодженням книг, документів, блокнотів тощо.

Слід зазначити, що асортименти виробів не є стабільною, постійною системою. Зміни у житті суспільства активно впливає формування асортименту і розробку нових видів виробів, на вдосконалення їх конструкцій.

Величезний вплив на асортимент має мода, обумовлюючи застосування нових способів декоративного оформлення, матеріалів та конструкцій. Спостерігається останні роки демократизація моди підказала ряд конструкцій сумок універсального характеру, сумок складаються, тобто. що застосовуються у різних умовах.

1.3. Деталі шкіргалантерейних виробів

Незалежно від виду та моделі в сумках розрізняють загальні для всіх деталі. Деталі сумок мають певне призначення, форму та розміри. Їх поділяють на зовнішні, внутрішні та проміжні.

Зовнішніми називають деталі, що знаходяться на зовнішній стороні виробу. Вони особливо різноманітні. Повторюючись у сумках різних моделей, ці деталі у кожному новому варіанті мають свої особливості.

Саме на них великий вплив надають зміни моди, зумовлюючи виключення тієї чи іншої деталі або поєднання її з іншою, а іноді й появою нових деталей. Форма та розміри зовнішніх деталей сумок можуть бути різними; вони залежать від призначення та розмірів виробу в цілому.

При конструюванні виробів зовнішні деталі сумок поділяють на основні та додаткові. До зовнішніх основних відносять деталі, що утворюють корпус виробу з передньої, задньої, нижньої та бічними сторонами, що визначають його розміри та форму, і є одночасно декоративними елементами. Це стінки, фальди, дно, клинчики та ботан.

У сумках із клапаном до зовнішніх основних деталей відносять також і клапан. Корпус сумки може бути утворений цілим полотном, яке у конструктивній єдності складає передню, задню, бічні та нижню частини сумок.

Стінки – деталі, що утворюють корпус виробу з передньою та задньою його сторонами. Звідси з'явилися назви передньої та задньої стінок. Частиною стінки можуть бути фальди та кокетка.

Ф а л ь д і – це деталі, що дозволяють збільшити обсяг виробу у верхній частині. До них прикріплюють застібку-блискавку або рамковий замок.

Форма та розмір стінок визначаються формою та розміром всього виробу. Сумки можуть бути прямокутного, трапеціподібного, овального чи фігурного силуетів.

Д н о – основна деталь, що знаходиться між стінками та утворює корпус виробу в нижній частині. Ширина дна залежить від розміру виробу шириною. Довжина дна переважно відповідає довжині сумки в нижній його частині.

Нижня частина сумки може складатися із задньої та передньої стінок, що з'єднуються по середній лінії дна. У деяких виробках задня, передня та нижня

частини сумок можуть бути утворені у конструктивній єдності однією деталлю або цілим полотном.

Клинчик – деталь, що утворює бічну частину корпусу сумки. Висота клинчика переважно відповідає висоті стінок: ширина у верхній частині визначає величину розкриття сумки. Клинчик надає бічній частині сумки форму прямокутника, овалу або трапеції.

Ботан – деталь, яка у конструктивній єдності утворює бічні та нижні частини виробу (нижній ботан), бічні та верхню частини (верхній ботан), а також бічні, верхню та нижню частини (круговий ботан) та, з'єднуючись зі стінками, складає корпус сумки.

Клапан – деталь для закривання верхньої частини корпусу сумки або кишень. Він може бути у вигляді самостійної деталі, разом із задньою стінкою або з полотном сумки. Зазначені зовнішні деталі бувають цільними та складовими, наприклад, ботан нерідко складається з двох деталей, що зшиваються посередині нижньої частини. Це сприяє кращому використанню матеріалів, виконанню відповідного змісту художника, створенню колірних та фактурних контрастів.

До зовнішніх додаткових відносять фурнітуру. Особливе місце у виробництві шкіргалантерейних виробів займає фурнітура. Застосування тієї чи іншої виду фурнітура безпосередньо з конструюванням виробу.

За призначенням фурнітуру поділяють такі види:

- для замикання виробів – замки, кнопки, застібки та ін.;
- для скріплення та з'єднання окремих деталей – заклепки, ручкотримачі, штифти, скріпки та ін.;
- для запобігання виробам від швидкого зносу – кути, накладки, вставки, окантовки та ін.;
- для зручності користування виробом – ручки.

Фурнітуру виготовляють переважно із чорних та кольорових металів, пластичних мас, іноді з дерева та інших матеріалів.

На металеву фурнітуру наносять захисні антикорозійні та декоративні покриття: нікелюють, хромують, латунують, міднять, анодують, оксидують і т.д.

У деяких випадках фурнітуру (рамкові замки, пряжки) або її окремі частини обтягують або обклеюють матеріалом, з якого виготовлені зовнішні деталі виробу. Однак фурнітура завжди повинна відповідати всім експлуатаційним та естетичним вимогам відповідно до свого призначення та з урахуванням виду, конструкції та розміру виробу, а також кольору матеріалу, що застосовується. При використанні у виробі різних видів фурнітури вони повинні бути однаковими за формою, кольором, видом декоративного та антикорозійного покриття.

До фурнітури, призначеної для закривання виробів, наприклад сумок, відносять рамкові та клапанні замки, застібки-блискавки та різні замикаючі пристрої (пряжки зі шпеньком, рамки, кобурні кнопки, декоративні гудзики та ін.). Ця фурнітура дуже різноманітна за конструкцією, розмірами, формою, способами кріплення, замикаючими пристроями та іншими ознаками.

Замикаючі пристрої рамкових замків також різноманітні. Це заціпні головки, накладки і складніші пристрої типу клапанних замків, які можуть бути розташовані на задній і обох рамках, стінці передньої рамки і т.д.

Клапанні замки відрізняються великою різноманітністю видів, форм, конструкцій та розмірів. На сумки з клапаном в основному ставлять замки наступних видів:

- звичайні клапанні, що складаються з корпусу та накладки із замикаючим механізмом у корпусі замка;
- цупферні, що складаються з скоби, що встановлюється на корпусі виробу, та накладки з розташованим у ній замикаючим механізмом;

- замки-вертушки, що складаються з корпусу замку з головкою, що повертається (вертається) і накладки з прорізом для головки найрізноманітніших форм;

- замки-уривки – приховані замки, що мають низькі та головки спеціальної конструкції.

До цього ж виду можна віднести:

- рукавичні та кобурні кнопки;
- пряжки (зі шпеньками) різних форм та розмірів;
- різні замикаючі пристрої: гудзики або милиці та петлі, банти, wegaths або горти та шлевки та ін.

Клапанні замки прикріплюють до виробів за допомогою розгинних ніжок (склямер), заклепок, цвяхів. При виготовленні сумок, валіз, папок як фурнітуру широко використовують застібки-блискавки різних розмірів. Застібки-блискавки, замки-уривки, кнопки рукавички застосовують для закривання сумок, а також внутрішніх і зовнішніх кишень.

Внутрішні деталі – це різні кишені та перегородки. Внутрішні кишені, призначені для розміщення та зберігання предметів, можуть бути заставними, прорізними, з клапаном, на гумці, на застібці-блискавці, відкритими тощо. Перегородки, перегородки – кишені, середники служать поділу внутрішньої частини сумки на відділення. Перегородки-кишені можуть бути відкритими або закритими. На клапан, застібку-блискавку, цупфер, кнопку тощо. Середник – це перегородка-кишеня, що закривається на рамковий замок.

Проміжними називають деталі, розташованими між зовнішніми та внутрішніми деталями. За призначенням проміжні деталі поділяють такі види.

Жорсткі проміжні деталі застосовують для зміцнення конструкції та надання їй жорсткості. Ці деталі можуть бути з картону, вініпласту та щільного паперу. Залежно від ступеня зміцнення конструкції виробу бувають м'якою, напівжорсткою та жорсткою конструкцією. У сумках напівжорсткої конструкції

жорсткі проміжні деталі застосовують для закріплення дна, стінок або клинчиків. У сумках жорсткої конструкції всі основні деталі зміцнюють проміжними жорсткими деталями. Крім того, жорсткі прокладки використовують для додаткових деталей (ручок, цупферів, накладок та ін.)

М'які проміжні деталі служать для ущільнення конструкції та створення опуклої поверхні виробів або окремих основних та додаткових деталей. Деталі можуть бути виготовлені з поролону, вати, ватину, нетканих матеріалів, байки, фланелі та ін. Їх застосовують разом із жорсткими проміжними деталями або без них. Проміжні деталі з тасьми, тканини, картону, паперу та шнура (у вигляді смужок, накладок під замки та ін.). Використовують для натягу основних деталей на картон, пристроювання застібки-блискавки, прикріплення рамкових замків і т.д.

Зовнішні, внутрішні та проміжні деталі викроюють із матеріалів для верху, підкладки та прокладки виробів. Матеріал верху виробу служить для оформлення виворітного боку деталей (підкладки, ручки, цупфера, горта, запряжника, ремня, і алого клапана), а також застосовується для окремих деталей (внутрішня кишеня, перегородка, окантовка, гаманець, футляр для дзеркала, папка-вкладиш, портмоне).

Рамкові замки в жіночих сумках можуть бути двох груп складності:

- замки першої групи складності – простіші, двостулкові, без паяння кутів та обтяжки шкірою, з нижньою заправкою матеріалу, з голівкою металевою, пластмасовою або дерев'яною, забарвленою або обтягнутою шкірою;
- замки другої групи складності – дво- або трьох-стулкові, з паянням кутів, з бічною або верхньою заправкою матеріалу або з середником, з голівкою та накладкою, прикрашеною гравіюванням або штампуванням з металу, пластмаси, рогу, або з дерева, пофарбованих або обтягнутих шкірою.

2. РОЗРОБЛЕННЯ МАТЕМАТИЧНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ДЕКОРАТИВНИХ ЕЛЕМЕНТІВ ДЛЯ ВИРОБІВ ШКІРГАЛАНТЕРЕЇ

Прикраси виробу відносяться до додаткових зовнішнім деталям, що не утворюють корпус виробу, а призначені для допоміжних цілей. Ці деталі надзвичайно різноманітні, до них пред'являють підвищені естетичні вимоги.

Часто ті чи інші додаткові деталі, їх вигляд, форма, розміри характеризують стиль оформлення виробів і напрям моди.

Розглянемо такі додаткові деталі як прикраси виробу, а саме: банти, квіти і т.д. Так як деталі прикрас будуть мати складну конфігурацію і їх зовнішній контур в більшості випадках неможливо описати аналітично, то ми будемо апроксимувати їх зовнішні контури з необхідною точністю багатокутниками.

Багатокутник може бути однозначно відображений, якщо ми будемо знати координати вершин цього багатокутника і порядок їх проходження. Будь вершина A_i багатокутника однозначно визначається її координатами (X_i, Y_i) в Декартовій системі координат.

2.1. Математичне забезпечення для автоматизованого проектування банту

Бант складатиметься з двох деталей. Для однозначного визначення параметрів кожної з деталей необхідно знати наступні параметри (рис.2.1): сторона А - A , сторона В - B , верхня ширина банта - ShV , нижня ширина банта - ShN .

Побудова деталі банта по її параметрах здійснюється наступним чином:

- Будуємо трапеції $A_1A_2A_3A_4$ і $B_1B_2B_3B_4$ (рис.2.2);
- визначаємо кут φ ;
- визначаємо радіуси r, R та координати точки $O_1(X_{O_1}, Y_{O_1})$;

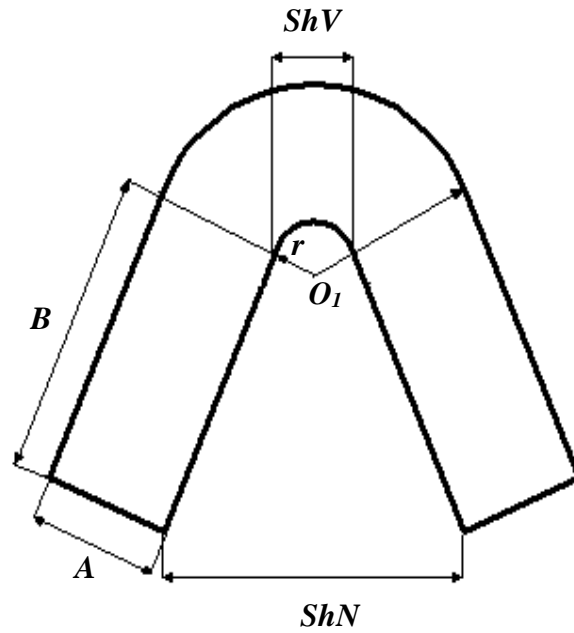
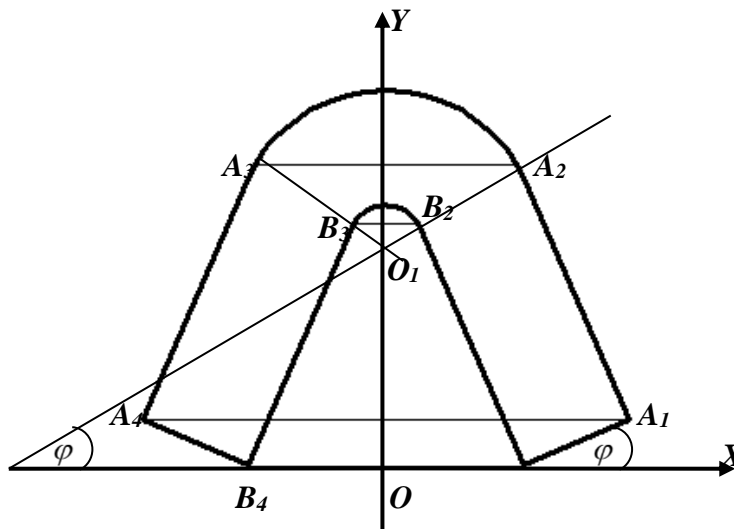


Рис. 2.1. Параметри, які однозначно визначають форму деталі



N n/n	X_{bi}	Y_{bi}
1	$ShN/2$	0
2	$ShV/2$	$\sqrt{B^2 - (ShN - ShV)^2}$
3	$-ShV/2$	$\sqrt{B^2 - (ShN - ShV)^2}$
4	$-ShN/2$	0
5	$ShN/2$	0

Рис. 2.2. Побудова деталі банту за її параметрами

Кут $\varphi = \frac{\pi}{2} - \angle B_4 B_1 B_2$. Так як $\operatorname{tg} \angle B_4 B_1 B_2 = \frac{\sqrt{B^2 - ((ShN - ShV)/2)^2}}{ShN - ShV}$, то

$$\varphi = \frac{\pi}{2} - \operatorname{arctg} \left(\frac{\sqrt{B^2 - ((ShN - ShV)/2)^2}}{ShN - ShV} \right). \quad (2.1)$$

Тоді координати вершин трапеції $A_1 A_2 A_3 A_4$ будуть визначатися наступним чином:

N n/n	Xa_i	Ya_i
1	$ShN/2 + A \cos \varphi$	$A \sin \varphi$
2	$ShV/2 + A \cos \varphi$	$A \sin \varphi + \sqrt{B^2 - (ShN - ShV)^2}$
3	$- ShV/2 - A \cos \varphi$	$A \sin \varphi + \sqrt{B^2 - (ShN - ShV)^2}$
4	$- ShN/2 - A \cos \varphi$	$A \sin \varphi$
5	$ShN/2$	$A \sin \varphi$

Радіуси r , R та координати точки $O_1(X_{O_1}, Y_{O_1})$ будуть визначатися наступним чином:

$$r = ShV/\cos \varphi; \quad R = r + A; \quad (2.2)$$

$$X_{O_1} = 0; \quad Y_{O_1} = \sqrt{B^2 - (ShN - ShV)^2} - r \cdot \sin \varphi. \quad (2.3)$$

Знаючи значення радіусів r , R та координат точки $O_1(X_{O_1}, Y_{O_1})$ легко описати дуги A_2A_3 та B_2B_3 :

$$\begin{aligned} XdA_i &= R \cdot \cos(\varphi + i \cdot \Delta\varphi) + X_{O_1} \\ YdA_i &= R \cdot \sin(\varphi + i \cdot \Delta\varphi) + Y_{O_1} \\ XdB_i &= r \cdot \cos(\varphi + i \cdot \Delta\varphi) + X_{O_1} \\ YdB_i &= r \cdot \sin(\varphi + i \cdot \Delta\varphi) + Y_{O_1} \end{aligned}, \quad \text{де} \quad \begin{aligned} i &= 0, 1, \dots, n \\ \Delta\varphi &= 2(\pi - \varphi)/n \end{aligned} \quad (2.4)$$

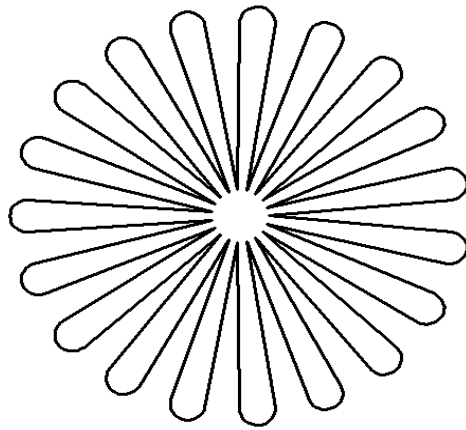
2.2. Математичне забезпечення для автоматизованого проектування квітів

Розглянемо проектування декоративних елементів у вигляді квітів на прикладах генерування деталей ромашки та троянди. Для проектування цих деталей необхідно знати наступні параметри деталі: Kl -кількість пелюсток, R -зовнішній радіус, r -внутрішній радіус, k ($k > 2$)-коефіцієнт. Приклади згенерованих ромашки і троянди представлені на рис 2.3.

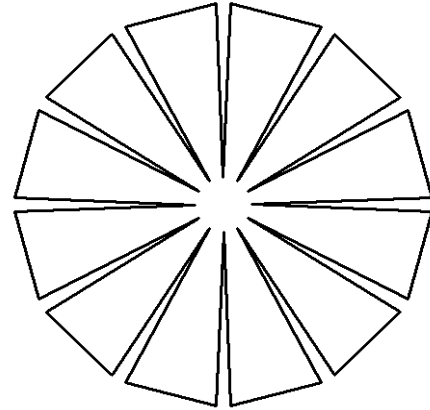
Для побудови квітки необхідно (рис. 2.4):

- побудувати правильні Kl -кутники (Kl -кількість пелюсток квітки), вписані в кола радіусів R і r ;

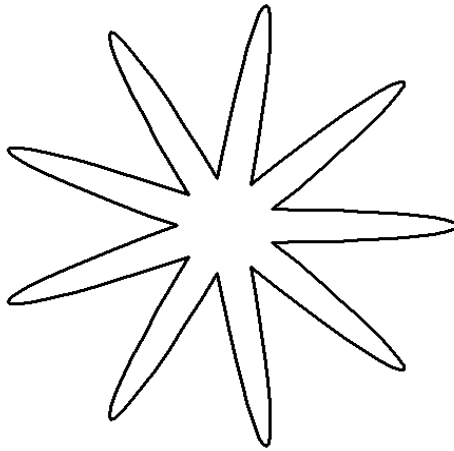
- визначити координати точок $B_i, i = 1, 2 \dots Kl$, які є середні точки відрізків $A_i A_{i+1}$;
- визначити r_0, d ;
- визначити кут φ ;
- побудувати дуги кіл радіусу r_0 , з центром в точці $B_i, i = 1, 2 \dots Kl$ з початковим кутом φ_i та кінцевим кутом $\varphi_i + \pi$.



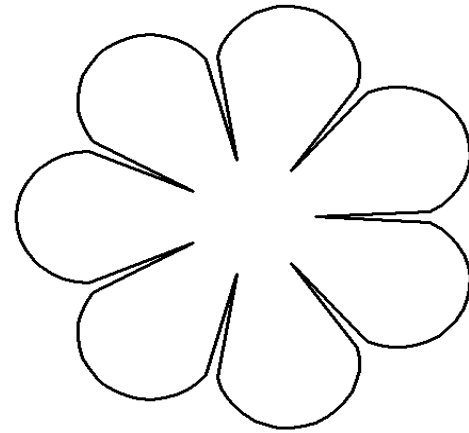
а



б



в



г

Рис. 2. 3. Проектування ромашки та троянди

а – ромашка_1(параметри: $Kl=19, R=45, r=6, k=4$)

б– ромашка_2(параметри: $Kl=12, R=45, r=6, k=14$)

в– ромашка_3(параметри: $Kl=9, R=45, r=10$)

г – троянда(параметри: $Kl=7, R=45, r=16, k=24$).

Для побудови правильних Kl -кутників (Kl -кількість пелюсток квітки), вписаних в кола радіусів R та r достатньо визначити координати вершин цих багатокутників:

$$\begin{aligned} Xa_i &= R \cdot \cos \beta_i \\ Ya_i &= R \cdot \sin \beta_i \\ Xc_i &= r \cdot \cos \beta_i \\ Yc_i &= r \cdot \sin \beta_i \end{aligned} \quad , \text{де} \quad \begin{aligned} i &= 1, 2, \dots, Kl \\ \beta_i &= 2\pi \cdot i / Kl \end{aligned} \quad (2.5)$$

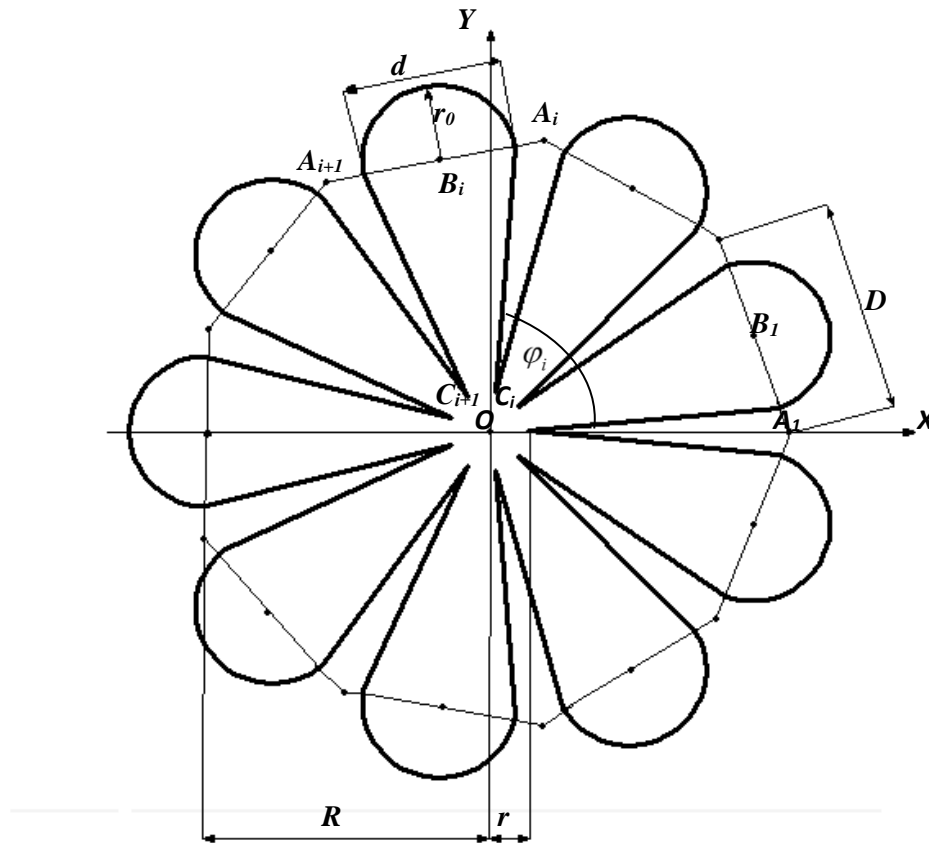


Рис. 2.4. Побудова креслення квітки

Координати точок B_i , $i=1, 2, \dots, Kl$ визначаються наступним чином:

$$\begin{aligned} Xb_i &= (Xa_i + Xa_{i+1}) / 2 \\ Yb_i &= (Ya_i + Ya_{i+1}) / 2 \end{aligned} \quad , \text{де} \quad \begin{aligned} i &= 1, 2, \dots, Kl \\ Xa_{kl+1} &= Xa_1 \\ Ya_{kl+1} &= Ya_1 \end{aligned} \quad (2.6)$$

Так як за визначенням $k = 2D / (D - d)$, то

$$d = \frac{D \cdot (k - 2)}{k} \quad (2.7)$$

Звідси
$$r_0 = d/2 = \frac{D \cdot (k-2)}{2k}. \quad (2.8)$$

Кут φ_i визначаються наступним чином: $\varphi_i = (i-1) \cdot \alpha + \alpha/k$, де $\alpha = 2\pi/Kl$.

Тоді i -у дугу квітки можна представити у вигляді многокутника, визначивши координати його вершин (Xd_{ij}, Yd_{ij}) , наступним чином:

$$\begin{aligned} Xd_{ij} &= r_0 \cdot \cos(\varphi_i + j \cdot \pi/N) + Xb_i, \\ Yd_{ij} &= r_0 \cdot \sin(\varphi_i + j \cdot \pi/N) + Yb_i, \end{aligned} \quad \text{де } \begin{matrix} i = 1, 2, \dots, Kl \\ j = 0, 1, \dots, N \end{matrix}. \quad (2.9)$$

2.3. Генерування одинарних декоративних елементів

Одинарні декоративні елементи, що входять до групового циклічного чи системного елементу мають форму квітки з пелюстками у вигляді: трикутника, трапеції, прямокутника, ромбу та краплі, лівої дуги кола, правої дуги кола, двох дуг кола. Приклади одинарних декоративних елементів у вигляді квітів представлені на рис. 2.5.

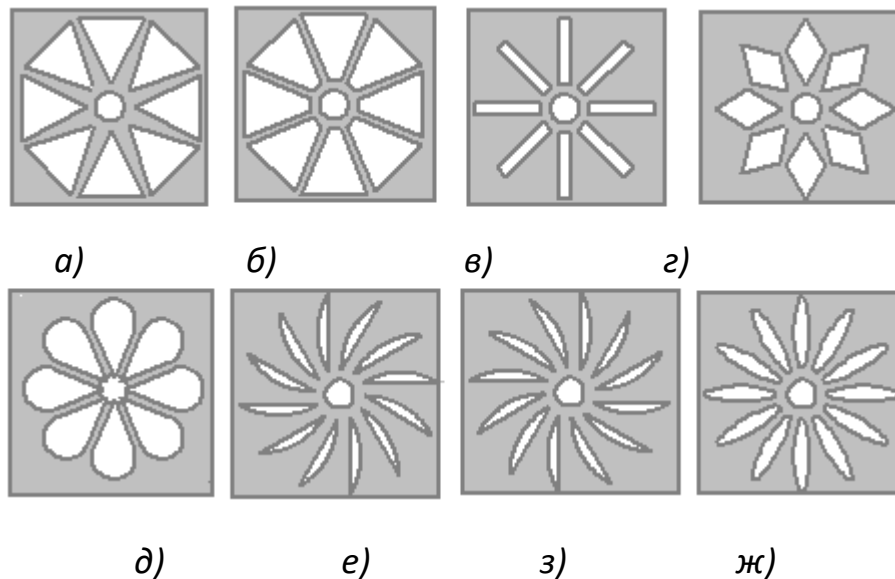


Рис. 2.5. Види одинарних декоративних елементів

- а) квітка з пелюстками у вигляді трикутника; б) квітка з пелюстками у вигляді трапеції; в) квітка з пелюстками у вигляді прямокутника; г) квітка з пелюстками у вигляді ромбу; д) квітка з пелюстками у вигляді краплі; е) квітка з пелюстками у вигляді лівої дуги кола; з) квітка з пелюстками у вигляді правої дуги кола; ж) квітка з пелюстками у вигляді двох дуг кола.

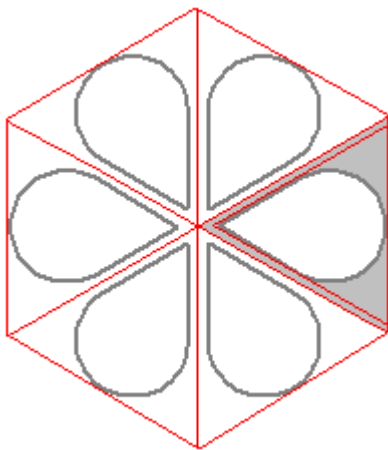
Маючи координати центрів одинарних декоративних елементів та радіуси кіл, що описані навколо них, можна перейти до генерування параметричної моделі для базового декоративного елементу одного із п'яти запропонованих варіантів. Для цього навколо базового декоративного елементу опишемо правильний багатокутник. Кількість сторін цього багатокутника визначається кількістю пелюстків у базового декоративного елементу (рис.2.6.а). Більш детально розглянемо зафарбований фрагмент декоративного елементу (рис. 2.6.б).

Нехай $OQ_1=OQ_2=R$. Тоді кут $\psi=\pi/N$, де N - кількість пелюстків у базового декоративного елементу та $a=|Q_1Q_2|=2 \cdot R \cdot \sin \psi$,

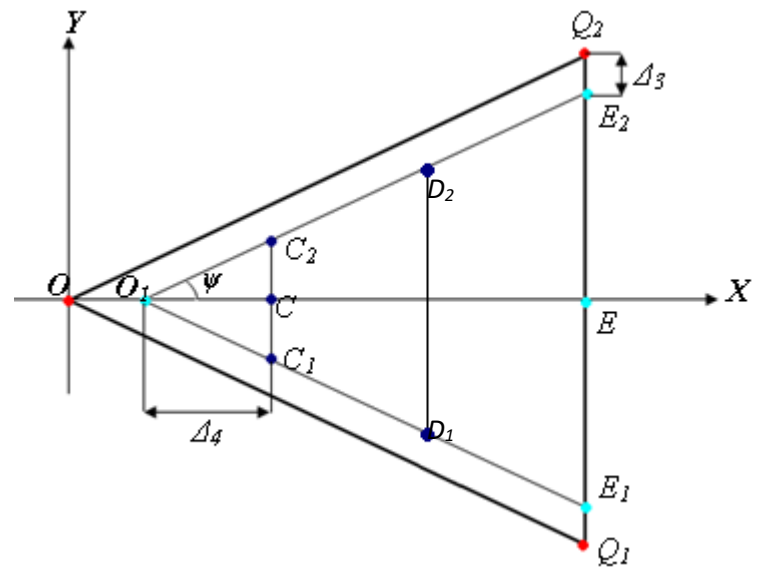
$$Q_1(X_{q1}, Y_{q1})=(R \cdot \cos \psi, -R \cdot \sin \psi) \text{ та } Q_2(X_{q2}, Y_{q2})=(R \cdot \cos \psi, R \cdot \sin \psi),$$

$$E_1(X_{e1}, Y_{e1})=(R \cdot \cos \psi, -R \cdot \sin \psi + \Delta_3) \text{ та } E_2(X_{e2}, Y_{e2})=(R \cdot \cos \psi, R \cdot \sin \psi - \Delta_3),$$

$$E(X_e, Y_e)=((X_{e1}+X_{e2})/2, (Y_{e1}+Y_{e2})/2)=(R \cdot \cos \psi, 0), O_1(X_{o1}, Y_{o1})=(\Delta_3 \cdot \operatorname{tg} \psi, 0).$$



а)



б)

Рис. 2.6. Вузлові точки, що визначають параметри одинарного декоративного елементу

З подібності ΔOQ_1Q_2 та $\Delta O_1C_1C_2$ легко визначити координати $C_1(X_{c1}, Y_{c1})$, $C_2(X_{c2}, Y_{c2})$, $C(X_c, Y_c)$, $D_1(X_{d1}, Y_{d1})$ та $D_2(X_{d2}, Y_{d2})$:

$$C_1(X_{c1}, Y_{c1}) = ((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a, -\Delta_4 \cdot \operatorname{tg} \psi);$$

$$C_2(X_{c2}, Y_{c2}) = ((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a, \Delta_4 \cdot \operatorname{tg} \psi);$$

$$C(X_c, Y_c) = ((X_{c1} + X_{c2}) / 2, (Y_{c1} + Y_{c2}) / 2) = ((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a, 0);$$

$$D_1(X_{d1}, Y_{d1}) = ((X_{c1} + X_{e1}) / 2, (Y_{c1} + Y_{e1}) / 2) =$$

$$= (((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a + R \cdot \cos \psi) / 2, (-\Delta_4 \cdot \operatorname{tg} \psi - R \cdot \sin \psi + \Delta_3) / 2);$$

$$D_2(X_{d2}, Y_{d2}) = ((X_{c2} + X_{e2}) / 2, (Y_{c2} + Y_{e2}) / 2) =$$

$$= (((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a + R \cdot \cos \psi) / 2, (\Delta_4 \cdot \operatorname{tg} \psi + R \cdot \sin \psi - \Delta_3) / 2)$$

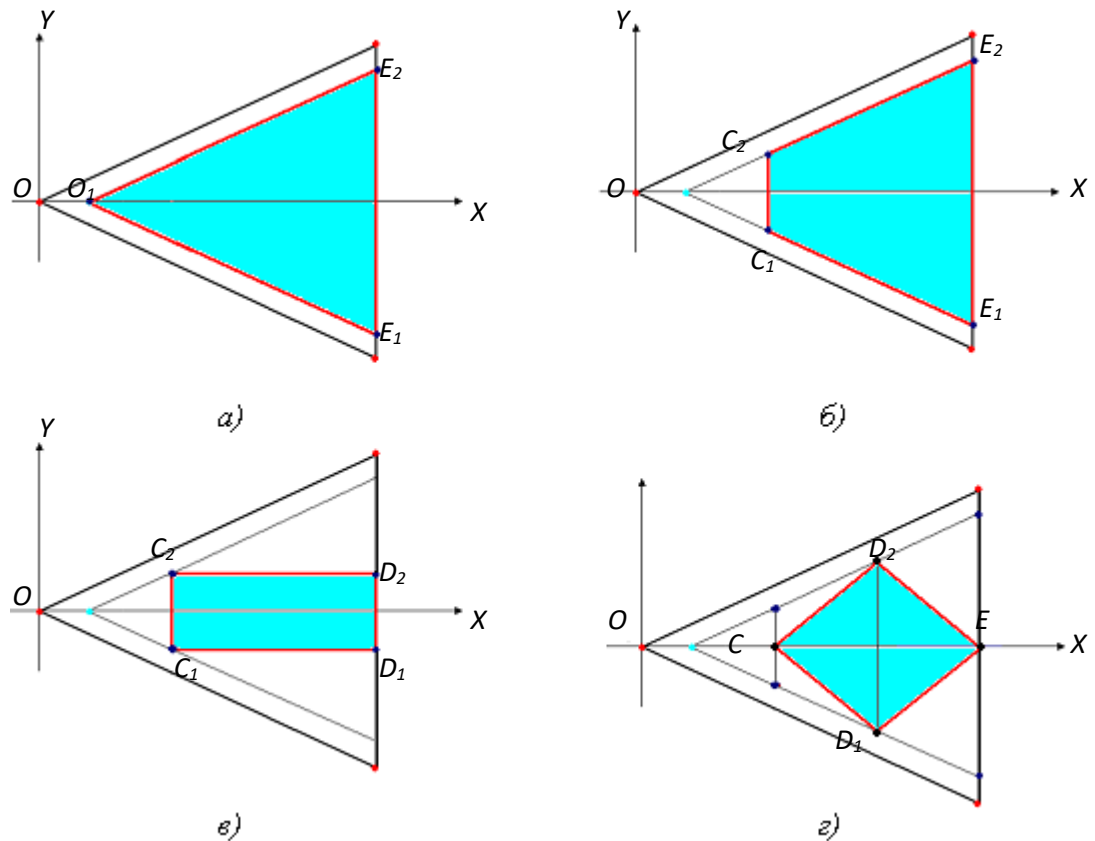


Рис. 2.7. Проектування пелюстків різної форми для декоративного елемента

Як очевидно із рис. 2.7 трикутна пелюстка описується $\Delta O_1E_1E_2$, трапецеїдальна пелюстка описується чотирикутником $C_1E_1E_2C_2$, прямокутна пелюстка описується чотирикутником $C_1D_1D_2C_2$, ромбовидна пелюстка описується чотирикутником CD_1ED_2 . Так як координати всіх цих вершин відомі, то параметричну модель однієї пелюстки представлених на рис. 2.7.a-г легко відтворити (таблиці 2.1-2.4), де $a=2 \cdot R \cdot \sin \psi$:

Таблиця 2.1

Параметрична модель пелюстки трикутної форми

$N n/n$	Xr	Yr
1	$\Delta_3 \cdot \operatorname{tg} \psi$	0
2	$R \cdot \cos \psi$	$-R \cdot \sin \psi + \Delta_3$
3	$R \cdot \cos \psi$	$R \cdot \sin \psi - \Delta_3$
4	$\Delta_3 \cdot \operatorname{tg} \psi$	0

Таблиця 2.2

Параметрична модель пелюстки трапецеїдальної форми

$N n/n$	Xr	Yr
1	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$-\Delta_4 \cdot \operatorname{tg} \psi$
2	$R \cdot \cos \psi$	$-R \cdot \sin \psi + \Delta_3$
3	$R \cdot \cos \psi$	$R \cdot \sin \psi - \Delta_3$
4	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$\Delta_4 \cdot \operatorname{tg} \psi$
5	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$-\Delta_4 \cdot \operatorname{tg} \psi$

Параметрична модель пелюстки прямокутної форми

$N n/n$	Xr	Yr
1	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$-\Delta_4 \cdot \operatorname{tg} \psi$
2	$R \cdot \cos \psi,$	$-\Delta_4 \cdot \operatorname{tg} \psi$
3	$R \cdot \cos \psi$	$\Delta_4 \cdot \operatorname{tg} \psi$
4	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$\Delta_4 \cdot \operatorname{tg} \psi$
5	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	$-\Delta_4 \cdot \operatorname{tg} \psi$

Окремо зупинимося на генеруванні пелюстки декоративного елемента у вигляді краплі (рис.2.8).

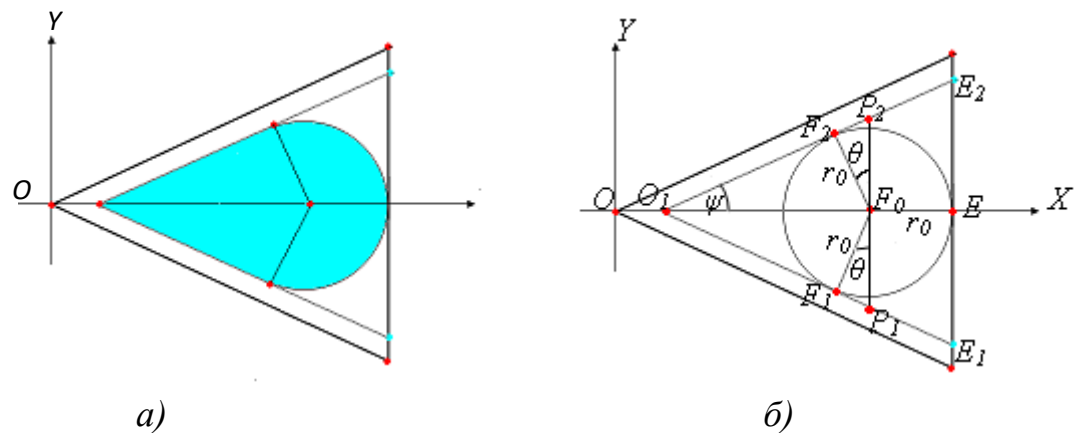


Рис. 2.8. Проектування пелюстків різної форми для декоративного елемента

Параметрична модель пелюстки ромбовидної форми

$N n/n$	Xr	Yr
1	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	0
2	$((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a + R \cdot \cos \psi) / 2$	$(-\Delta_4 \cdot \operatorname{tg} \psi - R \cdot \sin \psi + \Delta_3) / 2$

3	$R \cdot \cos \psi$	0
4	$((2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a + R \cdot \cos \psi) / 2$	$(\Delta_4 \cdot \operatorname{tg} \psi + R \cdot \sin \psi - \Delta_3) / 2$
5	$(2 \cdot \Delta_3 \cdot R \cdot \cos \psi + \Delta_4 \cdot a) / a$	0

Для отримання параметричної моделі пелюстки декоративного елемента у вигляді краплі необхідно мати координати центра $F_0(Xf_0, Yf_0)$ дуги F_1PF_2 , радіус дуги r_0 та початковий ω_1 та кінцевий ω_2 кути дуги.

Так як $\theta = \psi = \pi/N$, то $\omega_1 = -\pi/2 - \psi$ та $\omega_2 = \pi/2 + \psi$. Очевидно із рис.2.7.б, що центр шуканої дуги має наступні координати:

$$F_0(Xf_0, Yf_0) = (Xe - r_0, 0) = (R \cdot \cos \psi - r_0, 0).$$

Площа трикутника $E_1O_1E_2$ $S = |O_1E| \cdot |E_1E| / 2$. Так як $|E_1E| = R$ та $|O_1E| = R \cdot \sin 2\psi$, то $S = R^2 \cdot \sin 2\psi$. З другої сторони площа трикутника $E_1O_1E_2$ $S = p \cdot r_0$. Так як $p = (R + R + 2R \cdot \sin \psi) / 2 = R(1 + \sin \psi)$, то $S = R(1 + \sin \psi) \cdot r_0$. Звідси

$$r_0 = \frac{R \cdot \sin 2\psi}{2(1 + \sin \psi)}.$$

Так як дугу F_1PF_2 будемо представляти у вигляді ломаної лінії із заданою точністю ε , то для визначення кількості вершин Nr у апроксимуючої кривої скористаємося наступною нерівністю:

$$Nr \geq \frac{\omega_2 - \omega_1}{2 \operatorname{arc} \cos(1 - \frac{\varepsilon}{r_0})}.$$

Тоді параметричну модель базової пелюстки у вигляді краплі можна представити наступним чином:

$$\begin{aligned} Xr_0 &= \Delta_3 \cdot \operatorname{tg} \psi; & Yr_0 &= 0; \\ Xr_i &= r_0 \cdot \cos \omega_i + Xf_0; & Yr_i &= r_0 \cdot \sin \omega_i + Yf_0; & \text{де } \omega_i &= \omega_1 + (i-1) \cdot \frac{(\omega_2 - \omega_1)}{Nr} \\ Xr_{Nr+1} &= \Delta_3 \cdot \operatorname{tg} \psi; & Yr_{Nr+1} &= 0; \end{aligned}$$

Для отримання параметричної моделі пелюстки декоративного елемента у вигляді лівої дуги необхідно мати координати центра $D_3(Xd_3, Yd_3)$ дуги CD_6E , радіус дуги r_L та початковий ω_{1L} та кінцевий ω_{2L} кути дуги (рис. 2.9).

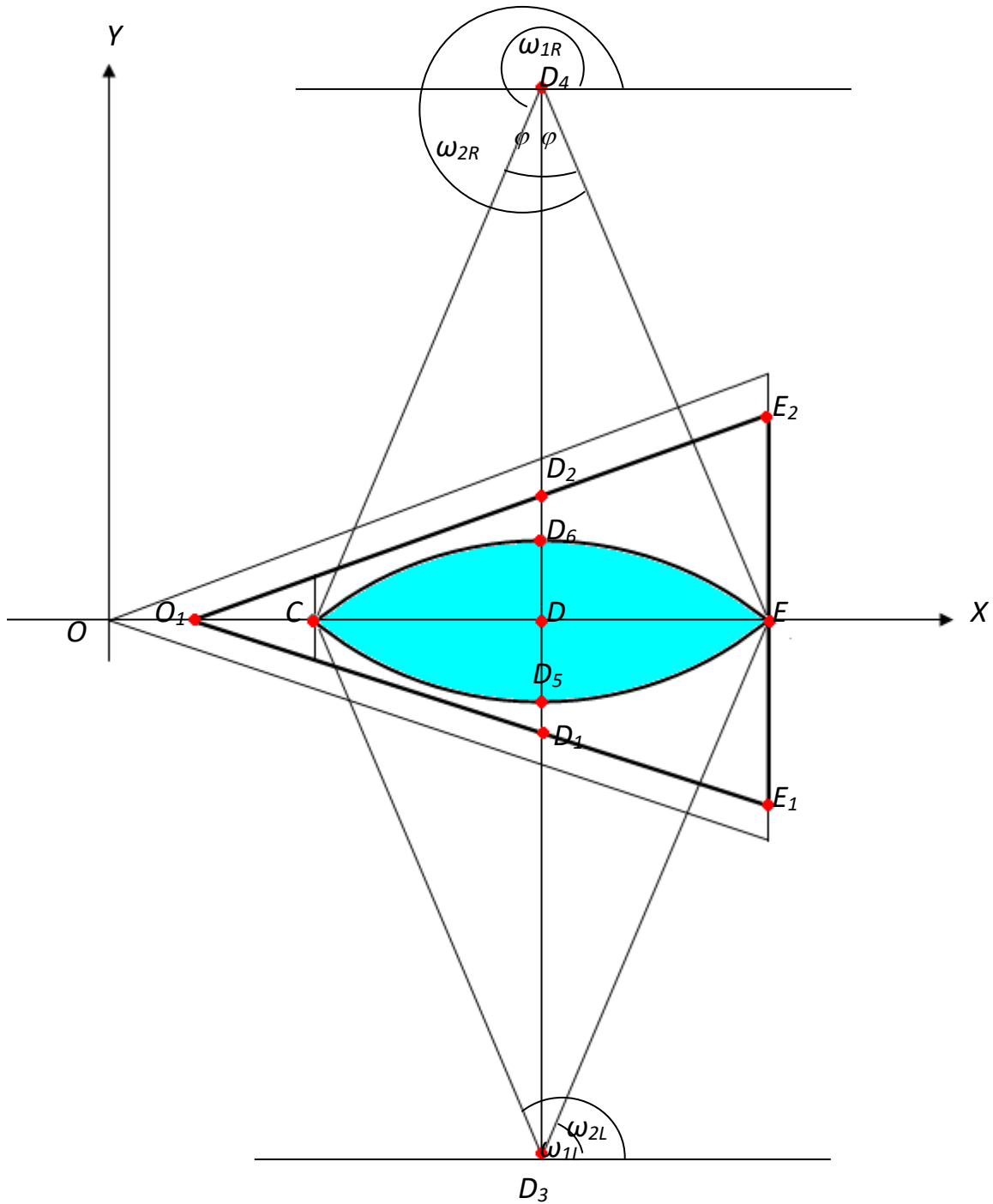


Рис. 2.9. Побудова пелюстки у вигляді дуг для одинарного декоративного елемента

Для отримання параметричної моделі пелюстки декоративного елемента у вигляді правої дуги необхідно мати координати центра $D_4(Xd_3, Yd_3)$ дуги CD_5E , радіус дуги r_R та початковий ω_{1R} та кінцевий ω_{2R} кути дуги (рис. 2.9). Для отримання параметричної моделі пелюстки декоративного елемента у вигляді лівої та правої дуги необхідно мати координати центра $D_3(Xd_3, Yd_3)$ лівої дуги CD_6E , радіус лівої дуги r_L та початковий ω_{1L} та кінцевий ω_{2L} кути лівої дуги, координати центра $D_4(Xd_3, Yd_3)$ правої дуги CD_5E , радіус правої дуги r_R та початковий ω_{1R} та кінцевий ω_{2R} кути правої дуги (рис. 2.8). Введемо позначення: $a=|CE|$, $h=|DD_5|=|DD_6|$, $r_L=|CD_3|=|ED_3|=|D_6D_3|$, $r_R=|CD_4|=|ED_4|=|D_4D_5|$. Із рис. 2.9 очевидно, що $r_L=r_R=r_0$, $|DE|=|CD|=a/2$ та $|D_4D|=|D_3D|=b$.

Визначимо довжину відрізка CE : $|CE| = \sqrt{(Xc - Xe)^2 + (Yc - Ye)^2}$. Прийmemo, що довжина відрізка D_6D_5 дорівнює 0.1-0.2 довжина відрізка CE . Із $\triangle DED_3$ за теоремою Піфагора маємо, що $|D_3E|^2 = |D_3D|^2 + |DE|^2$, або $r_0^2 = b^2 + a^2/4$.

Так як $r_0 = b + h$, то отримаємо $(b + h)^2 = b^2 + a^2/2$. Звідси отримаємо:

$$b = \frac{a^2 - 4h}{8h}, \quad r_0 = b + h = \frac{a^2 - 4h}{8h} + h, \quad \varphi = \arctg\left(\frac{4h \cdot a}{a^2 - 4h^2}\right).$$

Визначимо кути ω_{1L} , ω_{2L} , ω_{1R} та ω_{2R} (рис. 2.9):

$$\begin{aligned} \omega_{1L} &= \pi/2 - \varphi; & \omega_{2L} &= \omega_{1L} + 2\varphi; \\ \omega_{1R} &= 3\pi/2 - \varphi; & \omega_{2R} &= \omega_{1R} + 2\varphi. \end{aligned}$$

Тоді параметричну модель пелюстки декоративного елемента у вигляді лівої дуги можна представити наступним чином:

$$\begin{aligned} Xr_i &= r_0 \cdot \cos\left(\omega_{1L} + \frac{2\varphi}{n}i\right) + Xd_3 \\ Yr_i &= r_0 \cdot \sin\left(\omega_{1L} + \frac{2\varphi}{n}i\right) + Yd_3, \quad \text{де } i = 0, 1, \dots, n. \\ Xr_{n+1} &= Xr_0 & Yr_{n+1} &= Yr_0 \end{aligned}$$

Параметричну модель пелюстки декоративного елемента у вигляді правої дуги можна представити наступним чином:

$$\begin{aligned}
 Xr_i &= r_0 \cdot \cos(\omega_{1R} + \frac{2\varphi}{n}i) + Xd_4 \\
 Yr_i &= r_0 \cdot \sin(\omega_{1R} + \frac{2\varphi}{n}i) + Yd_4, \quad \text{де } i = 0, 1 \dots n. \\
 Xr_{n+1} &= Xr_0 \quad Yr_{n+1} = Yr_0
 \end{aligned}$$

Параметричну модель пелюстки декоративного елемента у вигляді лівої та правої дуги можна представити наступним чином:

$$\begin{aligned}
 Xr_i &= r_0 \cdot \cos(\omega_{1L} + \frac{2\varphi}{n}i) + Xd_3 \\
 Yr_i &= r_0 \cdot \sin(\omega_{1L} + \frac{2\varphi}{n}i) + Yd_3, \quad \text{де } i = 0, 1 \dots n. \\
 Xr_{n+i+1} &= r_0 \cdot \cos(\omega_{1R} + \frac{2\varphi}{n}i) + Xd_4 \\
 Yr_{n+i+1} &= r_0 \cdot \sin(\omega_{1R} + \frac{2\varphi}{n}i) + Yd_4
 \end{aligned}$$

Маючи параметричну модель базової пелюстки легко отримати параметричну модель j -ої пелюстки одинарного декоративного елемента:

$$\begin{aligned}
 X_{ji} &= Xr_i \cdot \cos \psi_j - Yr_i \cdot \sin \psi_j, \quad \text{де } i = 0, 1 \dots Nr \\
 Y_{ji} &= Xr_i \cdot \sin \psi_j + Yr_i \cdot \cos \psi_j, \quad \text{де } j = 1, 2 \dots N \\
 \psi_j &= 2 \cdot \psi \cdot (j - 1)
 \end{aligned}$$

3. РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ ДЕКОРАТИВНИХ ЕЛЕМЕНТІВ ДЛЯ ВИРОБІВ ШКІРГАЛАНТЕРЕЇ

3.1. Вибір системи програмування для практичної реалізації запропонованих методів та алгоритмів автоматизованого проектування декоративних елементів для виробів шкіргалантереї

Система програмування Delphi 7 фірми Enterprise (Borland) надає найбільш широкі можливості для програмування додатків ОС Windows.

Високопродуктивний інструмент візуального побудови додатків Delphi версії 7 включає в себе справжній компілятор коду і надає засоби візуального програмування, кілька схожі на ті, що можна виявити в Microsoft Visual Basic або в інших інструментах візуального проектування. В основі Delphi лежить мова Object Pascal, який є розширенням об'єктно-орієнтованої мови Pascal. У Delphi також входять локальний SQL-сервер, генератори звітів, бібліотеки візуальних компонентів, і інше, необхідне для того, щоб відчувати себе абсолютно впевненим при професійній розробці інформаційних систем або просто програм для Windows-середовища.

Насамперед Delphi призначений для професійних розробників, бажаючих дуже швидко розробляти додатки. Delphi виробляє невеликі за розмірами високоефективні виконувані модулі (.exe і .dll), тому в Delphi мають бути, перш за все, зацікавлені ті, хто розробляє продукти на продаж. З іншого боку невеликі за розмірами і швидко виконувані модулі означають, що вимоги до клієнтських робочих місцях істотно знижуються - це має важливе значення і для кінцевих користувачів.

Переваги Delphi в порівнянні з аналогічними програмними продуктами :

- Швидкість розробки додатку;
- Висока продуктивність розробленого додатка;
- Низькі вимоги розробленого додатка до ресурсів комп'ютера;

- Можливість розширення за рахунок вбудовування нових компонент та інструментів в середовище Delphi;
- Можливість розробки нових компонентів та інструментів використовуючи розробки Delphi (існуючі компоненти та інструменти доступні у вихідних кодах);
- Вдале опрацювання ієрархії об'єктів.

Система програмування Delphi розрахована на програмування різних додатків і надає велику кількість компонентів для цього. До того ж роботодавців цікавить, насамперед, швидкість і якість створення програм, а ці характеристики може забезпечити тільки середовище візуального проектування, здатне взяти на себе значні обсяги рутинної роботи з підготовки додатків, а також узгодити діяльність групи постановників, кодувальників, тестерів і технічних письменників. Можливості Delphi повністю відповідають подібним вимогам і підходять для створення систем будь-якої складності.

Для того щоб обгрунтувати, чому наш вибір зупинився на Borland Delphi 7, досить просто перерахувати деякі недоліки мови C ++ порівняно з ObjectPascal :

1. Треба робити багато ініціалізації (реєструвати клас вікна, організовувати цикл обробки повідомлень, створювати віконну функцію, піктограму та інше
2. Частково бути системним програмістом. На Delphi-ж системне програмування вже вбудовано і ініціалізація працює за замовчуванням, тому програміст головний акцент робить на своїх алгоритмах, а не на організації допоміжних робіт.
3. Значно більша, порівняно з Object Pascal, складність мови, навіть, незважаючи на компактність коду, виникають складнощі в його сприйнятті.
4. Одна особливість, на мій погляд, мови C ++ дуже псує цю мову - він чутливий до регістру символів, тобто змінна A і змінна a - це різні змінні.

5. У Delphi класи (об'єкти) можуть розташовуватися тільки в динамічній пам'яті, а C ++ у будь-якій пам'яті (статична, стек, динамічна). Це додає безпеки програмування в Delphi.

3.2. Опис основних процедур розробленого програмного продукту, які забезпечують генерування декоративних елементів для виробів шкіргалантереї

Процедура *GrIm1* буде креслення многокутника за наступними параметрами: *Nd*- кількість вершин многокутника, *Xd, Yd*- масиви з координатами вершин многокутника, *Xce, Yce*-координати центру області, в яку виводиться креслення многокутника, *mxy*- масштаб, в якому виводиться креслення многокутника.

```

Procedure GrIm1(Nd:integer;Xd,Yd:array of real; Xce,Yce,mxy:real);
  Var i:integer;
  Xr,Yr:array[0..4000] of integer;
  begin
    For i:=0 to Nd-1 do
      begin
        Xr[i]:=round(mxy*xd[i]+Xce);
        Yr[i]:=round(-mxy*yd[i]+Yce);
      end;
    // Form1.Image1.Picture:=Nil;
    With Form1.Image1.Canvas DO
      begin
        Pen.Width:=3;
        Pen.Color:=clBlack;
        MoveTo(Xr[0],Yr[0]);
        For i:=0 to Nd-1 do LineTo(Xr[i],Yr[i]);
      end;
    end;
  end;

```

Процедура *Det1* генерує зірку, елементами якої є прямокутники. Параметри зірки: *N* – кількість сторін; *Rz*- радіус описаного кола навколо зірки(зовнішній радіус), *Rv*- внутрішній радіус. В даному випадку *N:=10, Rz:=45, Rv=10*.

Procedure Det1;

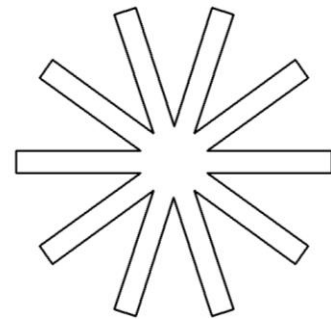
```

Var i:integer;
Xce,Yce:real;
begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rv:=5; Rz:=40; N:=14;A:=5;
  Fi:=2*Pi/N;
  Av:=2*Rv*sin(Fi/2);
  If A>Av then A:=Av;
  Az:=2*Rz*sin(Fi/2);
  TetaV:=A*Fi/Av;
  TetaZ:=A*Fi/Az;
  For i:=0 to N-1 do
    begin
      Xz[2*i]:=Rz*cos(i*Fi-TetaZ/2);
      Xz[2*i+1]:=Rz*cos(i*Fi+TetaZ/2);
      Yz[2*i]:=Rz*sin(i*Fi-TetaZ/2);
      Yz[2*i+1]:=Rz*sin(i*Fi+TetaZ/2);
      Xv[2*i]:=Rv*cos(i*Fi-TetaV/2);
      Xv[2*i+1]:=Rv*cos(i*Fi+TetaV/2);
      Yv[2*i]:=Rv*sin(i*Fi-TetaV/2);
      Yv[2*i+1]:=Rv*sin(i*Fi+TetaV/2);
    end;
  For i:=0 to N-1 do
    begin
      Xd[4*i]:=Xv[2*i];
      Xd[4*i+1]:=Xz[2*i];
      Xd[4*i+2]:=Xz[2*i+1];
      Xd[4*i+3]:=Xv[2*i+1];
      Yd[4*i]:=Yv[2*i];
      Yd[4*i+1]:=Yz[2*i];
      Yd[4*i+2]:=Yz[2*i+1];
      Yd[4*i+3]:=Yv[2*i+1];
    end;
  Xd[4*N]:=Xd[0];
  Yd[4*N]:=Yd[0];
  Nd:=4*N+1;
  Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
  GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

```

Вкажіть параметри деталі

Кількість сторін	<input type="text" value="10"/>
Зовнішній радіус	<input type="text" value="45"/>
Внутрішній радіус	<input type="text" value="10"/>
Ширина сторони	<input type="text" value="7"/>



Процедура *Det4* генерує зірку, елементами якої є півеліпси та трапеції. Параметри зірки: N – кількість сторін; Rz - радіус описаного кола навколо зірки(зовнішній радіус), A , B - відповідно півосі A , B півеліпсу. В даному випадку $N=10$, $Rz=45$, $A=10$, $B=25$.

```

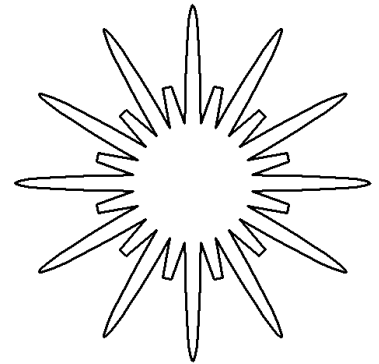
procedure Det4;
  Var i,Ne,j,k,p,q:integer;
  Xce,Yce,H,Alfa,Fi1,Av1, Az1,TetaV,TetaZ,Rv1:real;
  Xc,Yc,Xe,Ye:array[0..40]of real;
begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rz:=40; N:=6; A:=15;
  N:=N*2; Ne:=19;
  B:=30; Alfa:=Pi/(Ne-1);
  If B>=Rz then B:=Rz-1;
  Fi:=2*Pi/N;
  H:=Rz-B;
  Av:=H*sin(Fi/2)/cos(Fi/2);
  If A>Av then A:=Av;
  Rv:=Av/sin(Fi/2);
  For i:=0 to N do
    begin
      Xv[i]:=Rv*cos(i*Fi-Fi/2);
      Yv[i]:=Rv*sin(i*Fi-Fi/2);
    end;
  // GrIm1(N+1,Xv,Yv,Xce,Yce);
  For i:=0 to N-1 do
    begin
      Xc[i:]=(Xv[i]+Xv[i+1])/2;
      Yc[i:]=(Yv[i]+Yv[i+1])/2;
    end;
  // GrIm1(N,Xc,Yc,Xce,Yce);
  For i:=0 to Ne-1 do
    begin
      Xe[i]:=B*cos(i*Alfa-Pi/2);
      Ye[i]:=A*sin(i*Alfa-Pi/2);
    end;
  // A1:=5; Rz1:=20;
  p:=1;
  Fi1:=Fi/p;

  Rv1:=Rv;
  Av1:=2*Rv1*sin(Fi1/2);
  If A1>Av1 then A1:=Av1;
  Az1:=2*Rz*sin(Fi1/2);
  TetaV:=A1*Fi/Av1;
  TetaZ:=A1*Fi/Az1;
  q:=-1;
  For j:=0 to N-1 do
    begin
      if j mod 2=0 then

```

Вкажіть параметри деталі

Кількість сторін	<input type="text" value="12"/>
Зовнішній радіус	<input type="text" value="45"/>
Піввісь А	<input type="text" value="10"/>
Піввісь В	<input type="text" value="25"/>



```

For i:=0 to Ne-1 do
  begin
    q:=q+1;
    Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
    Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
  end;

if j mod 2=1 then
  begin
    Xz[0]:=Rz1*cos(j*Fi-TetaZ/2);
    Xz[1]:=Rz1*cos(j*Fi+TetaZ/2);
    Yz[0]:=Rz1*sin(j*Fi-TetaZ/2);
    Yz[1]:=Rz1*sin(j*Fi+TetaZ/2);
    Xv[0]:=Rv1*cos(j*Fi-TetaV/2);
    Xv[1]:=Rv1*cos(j*Fi+TetaV/2);
    Yv[0]:=Rv1*sin(j*Fi-TetaV/2);
    Yv[1]:=Rv1*sin(j*Fi+TetaV/2);
    q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
    q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
    q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
    q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
  end;
end;
Nd:=q+2;
Xd[Nd-1]:=Xd[0];
Yd[Nd-1]:=Yd[0];
Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

```

Процедура *Det7* генерує зірку, елементами якої є півеліпси та трикутники. Параметри зірки: N – кількість сторін; Rz - радіус описаного кола навколо зірки(зовнішній радіус), Rv - радіус описаного кола навколо трикутників(зовнішній радіус1), p - кількість трикутників між двома півеліпсами, A , B - відповідно півосі A , B півеліпсу. В даному випадку $N=7$, $Rz=40$, $Rv=35$, $A=15$, $B=30$, $p=3$.

```

procedure Det7;
Var i,Ne,j,k,q,m:integer;
Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,Rv1:real;
Xc,Yc,Xe,Ye:array[0..40]of real;
begin
  Xce:=Form1.Image1.Width/2;

```

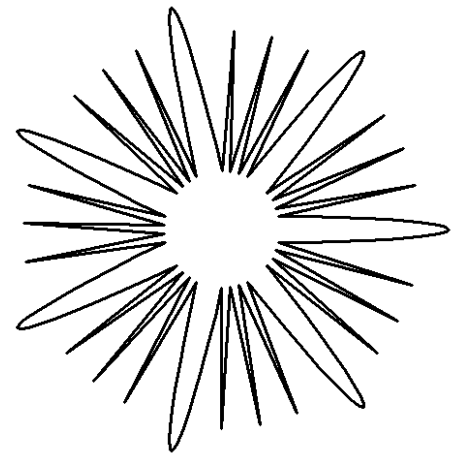
```

Yce:=Form1.Image1.Height/2;
// Rz:=40; N:=6; A:=15; B:=30;
N:=N*2; Ne:=19;
Alfa:=Pi/(Ne-1);
If B>=Rz then B:=Rz-1;
Fi:=2*Pi/N;
H:=Rz-B;
Av:=H*sin(Fi/2)/cos(Fi/2);
If A>Av then A:=Av;
Rv:=Av/sin(Fi/2);
For i:=0 to N do
begin
Xv[i]:=Rv*cos(i*Fi-Fi/2);
Yv[i]:=Rv*sin(i*Fi-Fi/2);
end;
// GrIm1(N+1,Xv,Yv,Xce,Yce);
For i:=0 to N-1 do
begin
Xc[i:]=(Xv[i]+Xv[i+1])/2;
Yc[i:]=(Yv[i]+Yv[i+1])/2;
end;
// GrIm1(N,Xc,Yc,Xce,Yce);
For i:=0 to Ne-1 do
begin
Xe[i]:=B*cos(i*Alfa-Pi/2);
Ye[i]:=A*sin(i*Alfa-Pi/2);
end;
//A1:=15; p:=3;
Fi1:=Fi/p;
// Rz1:=25;
Rv1:=Rv;
Av1:=2*Rv1*sin(Fi1/2);
A1:=Av1;
Az1:=2*Rz1*sin(Fi1/2);
TetaV:=A1*Fi1/Av1;
TetaZ:=A1*Fi1/Az1;
q:=-1;
For j:=0 to N-1 do
begin
if j mod 2=0 then
For i:=0 to Ne-1 do
begin
q:=q+1;
Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
end;
if j mod 2=1 then

```

Вкажіть параметри деталі

Кількість сторін	<input type="text" value="7"/>
Зовнішній радіус	<input type="text" value="40"/>
Піввісь А	<input type="text" value="15"/>
Піввісь В	<input type="text" value="30"/>
Зовнішній радіус 1	<input type="text" value="35"/>
Кількість ланцюгів	<input type="text" value="3"/>



```

for m:=0 to p-1 do
begin
  Xz[0]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
  Xz[1]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
  Yz[0]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
  Yz[1]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
  Xv[0]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m));
  Xv[1]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m+1));
  Yv[0]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m));
  Yv[1]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m+1));
  q:=q+1; Xd[q]:=Xv[0];      Yd[q]:=Yv[0];
  q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2;  Yd[q]:=(Yz[0]+Yz[1])/2;
  q:=q+1; Xd[q]:=Xv[1];      Yd[q]:=Yv[1];
end;
end;
Nd:=q+2;
Xd[Nd-1]:=Xd[0];
Yd[Nd-1]:=Yd[0];

Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

```

Процедура *Det9* генерує зірку, елементами якої є трикутники. Параметри зірки: N – кількість сторін; $Rz1$ - радіус описаного кола навколо зірки(зовнішній радіус1), $Rz2$ - радіус описаного кола навколо менших трикутників(зовнішній радіус2), Rv – внутрішній радіус. В даному випадку $N=10$, $Rz1=45$, $Rz2=30$, $Rv=8$.

```

procedure Det9;
Var i,q:integer;
Xce,Yce,TetaZ1,TetaZ2,Az1,Az2:real;
begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rv:=5; Rz1:=40; Rz2:=20; N:=6;
  N:=N*2;
  Fi:=2*Pi/N;
  Av:=2*Rv*sin(Fi/2);
  A:=Av;
  Az1:=2*Rz1*sin(Fi/2);
  Az2:=2*Rz2*sin(Fi/2);
  TetaV:=A*Fi/Av;
  TetaZ1:=A*Fi/Az1;
  TetaZ2:=A*Fi/Az2;
  q:=-1;
  For i:=0 to N-1 do

```

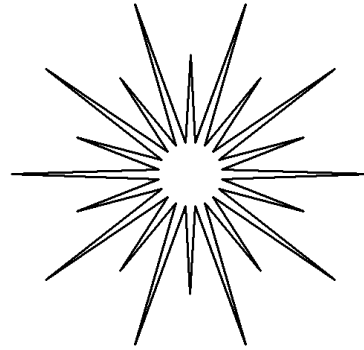
Вкажіть параметри деталі	
Кількість сторін	10
Зовнішній радіус 1	45
Зовнішній радіус 2	30
Внутрішній радіус	8
Ввести	

```

begin
  if i mod 2=0 then
    begin
      TetaZ:=TetaZ1;
      Rz:=Rz1;
    end
  else
    begin
      TetaZ:=TetaZ2;
      Rz:=Rz2;
    end ;
  Xz[0]:=Rz*cos(i*Fi-TetaZ/2);
  Xz[1]:=Rz*cos(i*Fi+TetaZ/2);
  Yz[0]:=Rz*sin(i*Fi-TetaZ/2);
  Yz[1]:=Rz*sin(i*Fi+TetaZ/2);
  Xv[0]:=Rv*cos(i*Fi-TetaV/2);
  Xv[1]:=Rv*cos(i*Fi+TetaV/2);
  Yv[0]:=Rv*sin(i*Fi-TetaV/2);
  Yv[1]:=Rv*sin(i*Fi+TetaV/2);

  q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
  q:=q+1; Xd[q:]=(Xz[0]+Xz[1])/2; Yd[q:]=(Yz[0]+Yz[1])/2;
  q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
end;
q:=q+1;
Xd[q]:=Xd[0];
Yd[q]:=Yd[0];
Nd:=q+1;
Scala( Nd, Xd, Yd, Form1.Image1.Width-40, Form1.Image1.Height-40, mxy);
GrIm1(Nd, Xd, Yd, Xce, Yce, mxy);
end;

```



Процедура *Det13* генерує зірку, елементами якої є прямокутники та трикутники. Параметри зірки: N – кількість сторін; Rz - радіус описаного кола навколо зірки(зовнішній радіус1), Rz - радіус описаного кола навколо трикутників(зовнішній радіус2), Rv – внутрішній радіус, p - кількість трикутників між двома сусідніми прямокутниками. В даному випадку $N=8$, $Rz=45$, $Rz1=25$, $Rv=10$, $p=3$.

```

procedure Det13;
Var i,Ne,j,k,q,m,Nd:integer;
Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaV1,TetaZ,TetaZ1,Rv1:real;
Xc,Yc,Xe,Ye:array[0..40]of real;

```



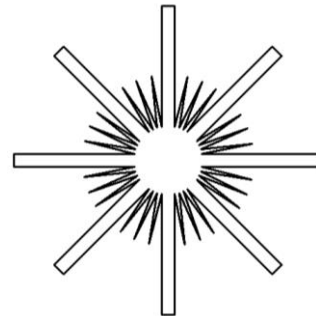
```

begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rz:=40; N:=6;
  N:=N*2;
  //A:=15; Ne:=19;
  // B:=30; Alfa:=Pi/(Ne-1);
  //If B>=Rz then B:=Rz-1;
  Fi:=2*Pi/N;
  // H:=Rz-B;
  // H:=Rv*cos(Fi/2);
  //Av:=H*sin(Fi/2)/cos(Fi/2);
  Av:=2*Rv*sin(Fi/2);
  // If A>Av then
  A:=Av;
  Az:=2*Rz*sin(Fi/2);
  TetaZ:=A*Fi/Az;
  TetaV:=A*Fi/Av;
  // Rv:=Av/sin(Fi/2);
  //A1:=15; p:=3;
  Fi1:=Fi/p;
  // Rz1:=25;
  Rv1:=Rv;
  Av1:=2*Rv1*sin(Fi1/2);
  // If A1>Av1 then
  A1:=Av1;
  Az1:=2*Rz1*sin(Fi1/2);
  TetaV1:=A1*Fi1/Av1;
  TetaZ1:=A1*Fi1/Az1;
  q:=-1;
  For j:=0 to N-1 do
    begin
      if j mod 2=0 then
        begin
          Xz[0]:=Rz*cos(j*Fi-TetaZ/2);
          Xz[1]:=Rz*cos(j*Fi+TetaZ/2);
          Yz[0]:=Rz*sin(j*Fi-TetaZ/2);
          Yz[1]:=Rz*sin(j*Fi+TetaZ/2);
          Xv[0]:=Rv*cos(j*Fi-TetaV/2);
          Xv[1]:=Rv*cos(j*Fi+TetaV/2);
          Yv[0]:=Rv*sin(j*Fi-TetaV/2);
          Yv[1]:=Rv*sin(j*Fi+TetaV/2);
          q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
          q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
          q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
          q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
        end;
      if j mod 2=1 then

```

Вкажіть параметри деталі:

Кількість сторін	<input type="text" value="8"/>
Зовнішній радіус 1	<input type="text" value="45"/>
Зовнішній радіус 2	<input type="text" value="25"/>
Внутрішній радіус	<input type="text" value="10"/>
Кількість ланцюгів	<input type="text" value="3"/>



```

for m:=0 to p-1 do
begin
  Xz[0]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+1));
  Xz[1]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+2));
  Yz[0]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+1));
  Yz[1]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+2));
  Xv[0]:=Rv1*cos(j*Fi-p*TetaV1/2+TetaV1*(m));
  Xv[1]:=Rv1*cos(j*Fi-p*TetaV1/2+TetaV1*(m+1));
  Yv[0]:=Rv1*sin(j*Fi-p*TetaV1/2+TetaV1*(m));
  Yv[1]:=Rv1*sin(j*Fi-p*TetaV1/2+TetaV1*(m+1));
  q:=q+1; Xd[q]:=Xv[0];      Yd[q]:=Yv[0];
  q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2;  Yd[q]:=(Yz[0]+Yz[1])/2;
  q:=q+1; Xd[q]:=Xv[1];      Yd[q]:=Yv[1];
end;
end;
Nd:=q+2;
Xd[Nd-1]:=Xd[0];
Yd[Nd-1]:=Yd[0];
Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

```

Процедура *Det15* генерує зірку, елементами якої є трапеції. Параметри зірки: N – кількість сторін; R_z - радіус описаного кола навколо зірки(зовнішній радіус1), R_v – внутрішній радіус, $k(k>2)$ – коефіцієнт розхилу. В даному випадку $N=6$, $R_z=45$, $R_v=6$, $k=4$.

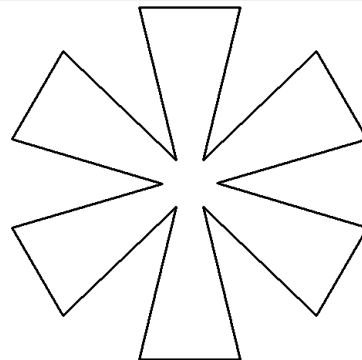
```

procedure Det15;
Var A,Fi,Xce,Yce,Xc,Yc:real;
    i,k,q:integer;
    Ydz,Xdz,Ydv,Xdv:array[0..50]of real;
begin
  // N:=9; Rz:=40; Rv:=5; k:=25;
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  Fi:=2*Pi/N;
  N:=N+1;
  For i:=0 to N-1 do
  begin
    Xdz[i]:=Rz*cos(i*Fi);
    Ydz[i]:=Rz*sin(i*Fi);
  end;
  For i:=0 to N-1 do
  begin
    Xdv[i]:=Rv*cos(i*Fi);
    Ydv[i]:=Rv*sin(i*Fi);
  end;

```

Вкажіть параметри деталі

Кількість сторін	<input type="text" value="6"/>
Зовнішній радіус	<input type="text" value="45"/>
Внутрішній радіус	<input type="text" value="6"/>
Коеф. розхилу(>2)	<input type="text" value="4"/>



```

end;
q:=-1;
For i:=0 to N-2 do
begin
q:=q+1; Xd[q]:=Xdv[i]; Yd[q]:=Ydv[i];
q:=q+1; NewPoint(Xdz[i],Ydz[i],Xdz[i+1],Ydz[i+1], p,Xd[q],Yd[q]);
q:=q+1; NewPoint(Xdz[i+1],Ydz[i+1],Xdz[i],Ydz[i], p,Xd[q],Yd[q]);
end;
q:=q+1; Xd[q]:=Xd[0]; Yd[q]:=Yd[0];
Nd:=q+1;
Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

```

3.4. Інструкції по роботі з програмним продуктом

Початок роботи з програмою розпочинається з запуску файлу *PrKonetsky.exe*. При цьому на екрані з'являється головне вікно програми прийме наступний вигляд(рис. 3.1).

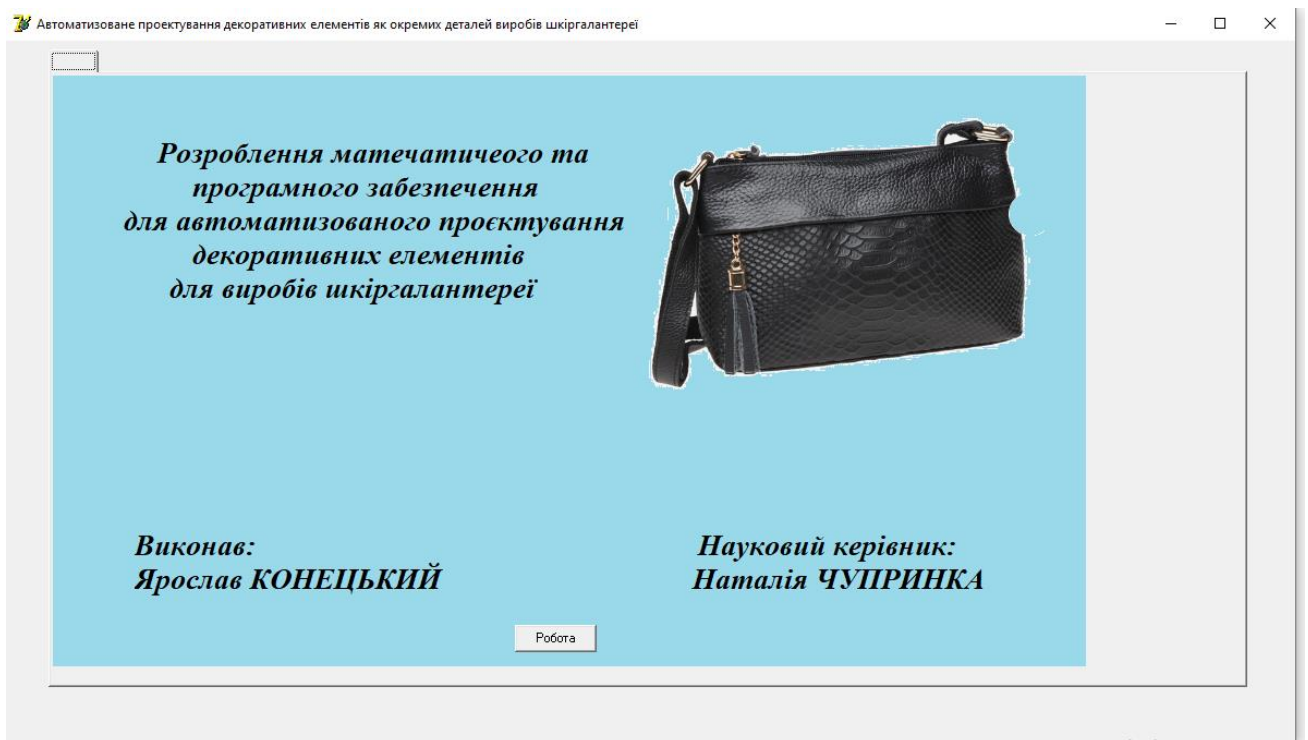


Рис. 3.1

Після натиску на кнопку «Робота» необхідно вибрати файл *.bmp з кресленнями необхідних плоских геометричних об'єктів та натиснути на кнопку «Ввести». (рис. 3.2).

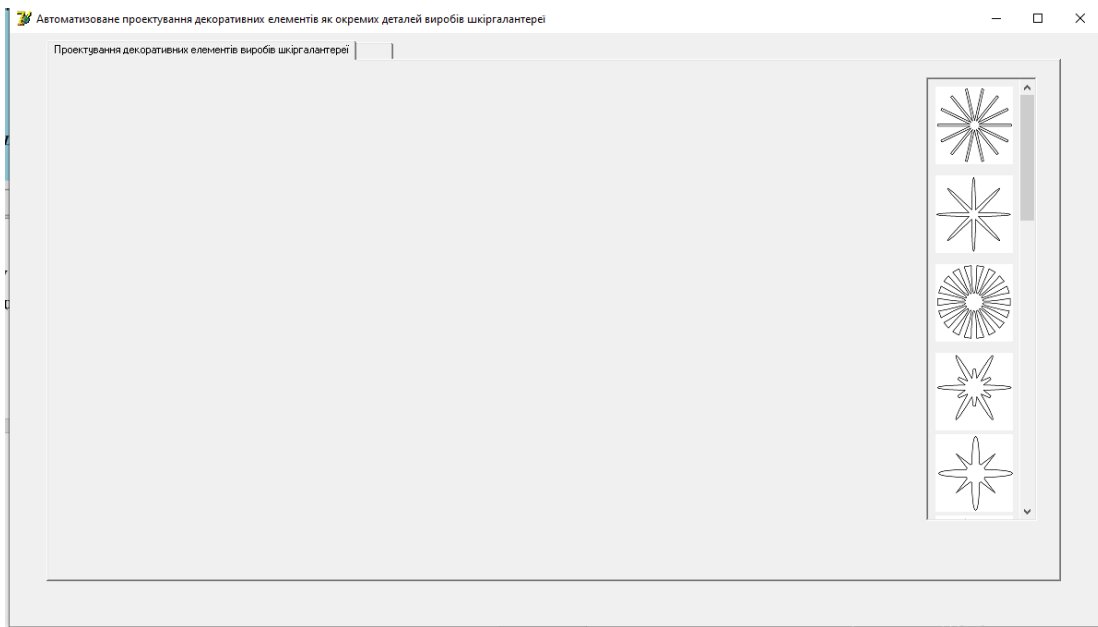


Рис. 3.2

Далі для підготовки інформації про зовнішні контури декоративного елемента необхідно вибрати його форму із меню, що розміщується ліворуч. Нехай це буде зірка, сторони якої будуть прямокутники. Підведемо курсор миші до неї в меню та натиснемо на кнопку миші. Головне вікно програми прийме наступний вигляд(рис. 3.3).

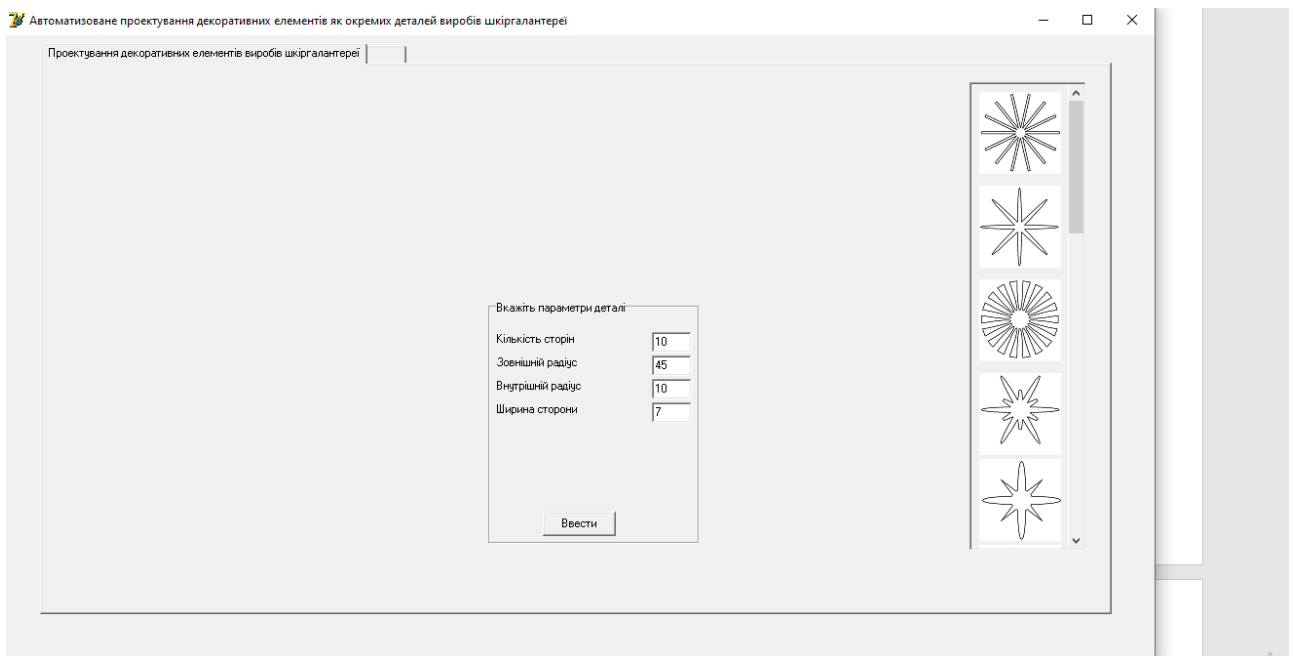


Рис. 3.3.

Після введення інформації, що необхідна для її однозначного відображення декоративного елемента, та натиску на кнопку «Ввести» в головному вікні програми з'явиться креслення спроектованого декоративного елемента (рис. 3.4). Аналогічно поступаємо при проектуванні будь-якого за формою із сімнадцяти запропонованих декоративних елементів, що містяться в меню.

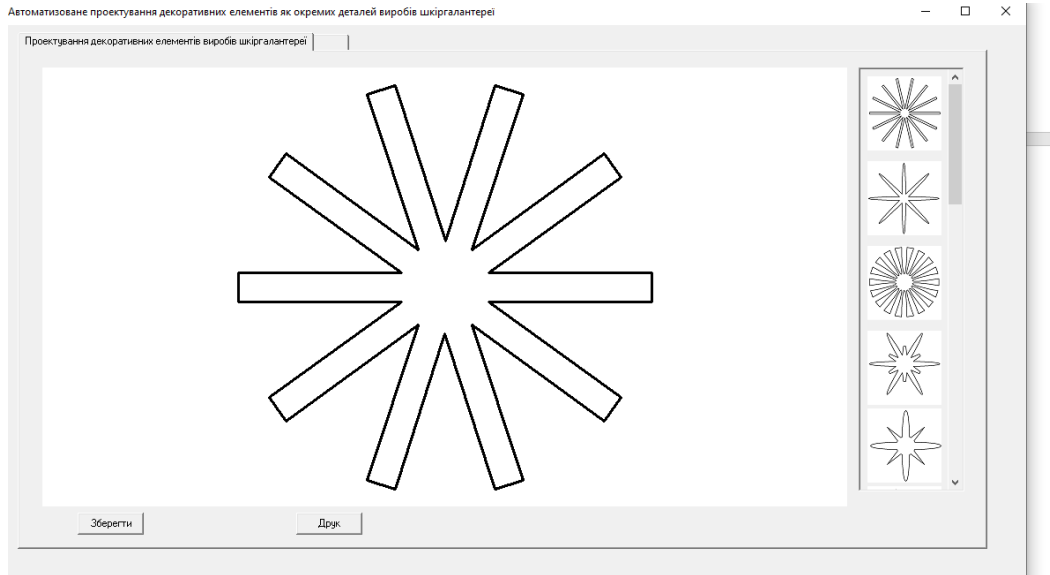


Рис. 3.4

Креслення спроектованого декоративного елемента можна зберегти на друк. Для цього необхідно натиснути на кнопку «Друк». Головне вікно програми прийме наступний вигляд(рис. 3.5).

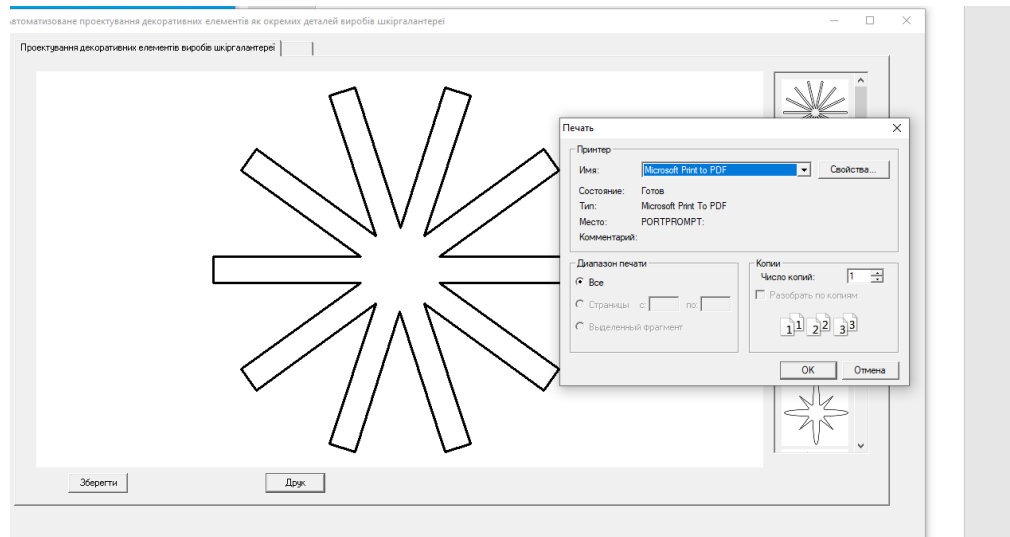


Рис. 3.5

Креслення спроектованого декоративного елемента представлено на рис. 3.6, де окрім креслення декоративного елемента визначається його площа S та периметер P і вказується у якому масштабі виведене це креслення(рис. 3.6).

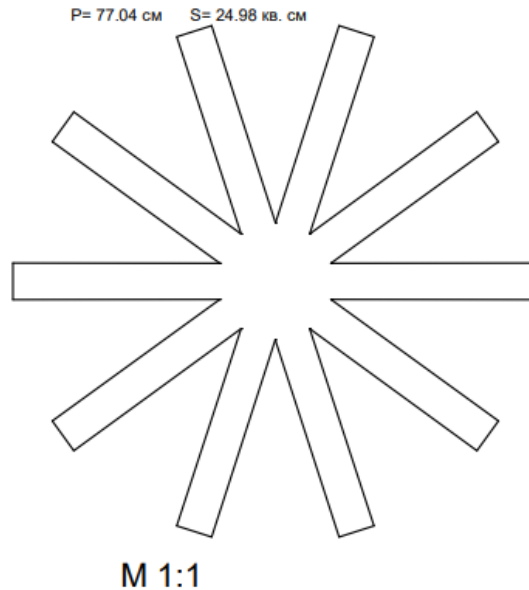


Рис. 3.6

Інформацію про зовнішній контур декоративного елемента, що спроектований, можна зберегти у файлі *.dgt. Для цього необхідно Для цього необхідно натиснути на кнопку «Зберегти» та вказати імя, в якому буде збережена інформація про про зовнішній контур декоративного елемента. Головне вікно програми прийме наступний вигляд(рис. 3.7).

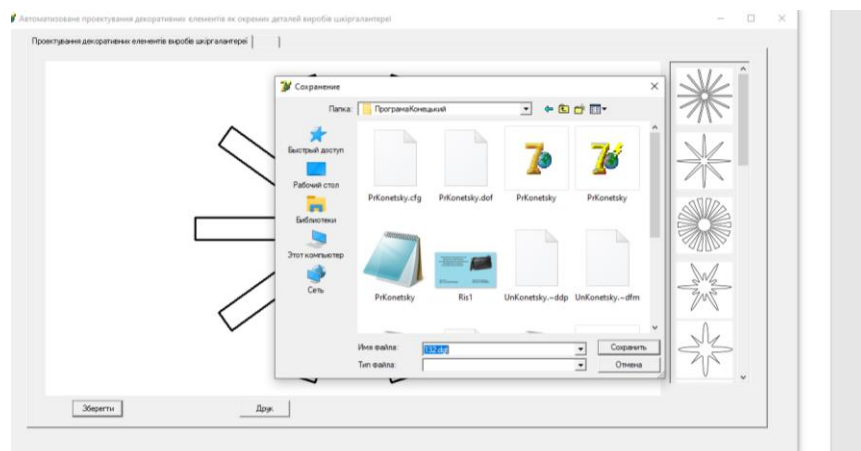


Рис. 3.7

Структура файлу *.dgt зі збереженою інформацією про зовнішній контур спроектованого декоративного елемента представлена нижче:

AAAAAAAA – назва моделі

gggggggg – службова інформація

1 – кількість декоративних елементів у моделі

Прикраса 1 – назва першого декоративного елемента у файлі з інформацією про зовнішні контури деталей

41 – кількість вершин на зовнішньому контурі декоративного елемента з назвою Прикраса 1

9.51	-3.09	Прикраса 1
44.89	-3.14	
44.89	3.14	
9.51	3.09	
9.51	3.09	
38.16	23.85	
34.47	28.93	
5.88	8.09	
5.88	8.09	
16.86	41.72	
10.89	43.66	
0.00	10.00	
0.00	10.00	
-10.89	43.66	
-16.86	41.72	
-5.88	8.09	
-5.88	8.09	
-34.47	28.93	
-38.16	23.85	
-9.51	3.09	
-9.51	3.09	
-44.89	3.14	
-44.89	-3.14	
-9.51	-3.09	
-9.51	-3.09	
-38.16	-23.85	
-34.47	-28.93	
-5.88	-8.09	
-5.88	-8.09	
-16.86	-41.72	
-10.89	-43.66	
-0.00	-10.00	

Відповідно координати
вершин X та Y на
зовнішньому контурі
декоративного елемента
з назвою Прикраса 1

-0.00	-10.00
10.89	-43.66
16.86	-41.72
5.88	-8.09
5.88	-8.09
34.47	-28.93
38.16	-23.85
9.51	-3.09
9.51	-3.09

*Відповідно координати
вершин X та Y на
зовнішньому контурі
декоративного елемента
з назвою Прикраса 1*

ВИСНОВКИ

Проведено дослідження та розроблені алгоритми для автоматизованого генерування декоративних елементів для виробів шкіргалантереї.

Запропоноване математичне забезпечення для генерування декоративних елементів для виробів шкіргалантереї реалізоване в прогрограмний продукт.

Розроблений програмний продукт дозволяє генерувати деталі виробів шкіргалантереї із комбінації простих геометричних об'єктів.

Цей програмний продукт легкий в користуванні, має дружній та привабливий інтерфейс. Може бути застосована на підприємствах в шкіргалантерейній галузі легкої промисловості.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вуштей О. А. Конструювання шкіргалантерейних виробів із використанням САПР / О. А. Вуштей, Н. В. Чупринка // Вісник КНУТД, – 2012. №2. – С. 13-16.
2. Чупринка Н. В. Автоматизоване проектування деталей жіночих сумок з використанням стандартних елементів / Н. В. Чупринка, С. С. Гаркавенко // Вісник КНУТД, – 2014. №6. – С. 42-48.
3. Чупринка Н. В. Параметрична модель циклічних групових декоративних елементів на деталях жіночих сумок / Н. В. Чупринка, С. С. Гаркавенко // Вісник КНУТД, – 2015. №3. – С. 20-27.
4. Чупринка Н. В. Автоматизоване проектування декоративних елементів на зовнішніх контурах деталей жіночих сумок / Н. В. Чупринка, С. С. Гаркавенко // Вісник КНУТД, – 2015. №4. – С. 237-242.
5. Чупринка Н. В. Автоматизована підготовка інформації про деталі шкіргалантерейних виробів / Н. В. Чупринка, С. С. Гаркавенко // Проблеми легкої промисловості України 2012, №2(20). – С. 65-69.
6. Чупринка Н. В. Автоматизоване проектування декоративних елементів жіночих сумок / Н. В. Чупринка, С. С. Гаркавенко // Легка промисловість, – 2014. – №3-4. – С. 46-47.
7. Дроменко В. І. Метод автоматизованого проектування рукавичкових виробів / В. І. Дроменко, Н. В. Чупринка // Тези доповідей X Всеукраїнської наукової конференції молодих вчених та студентів, 19-20 квітня 2011 р. „Наукові розробки молоді на сучасному етапі”. – Т2 - К.:КНУТД, 2011. – С. 58.
8. Чупринка Н. В. Автоматизоване проектування деталей шкіргалантерейних виробів / Н. В. Чупринка // Тези доповідей XI Всеукраїнської наукової конференції молодих вчених та студентів, 19-20 квітня „Наукові розробки молоді на сучасному етапі”. – Т2 - К.:КНУТД, 2012. – С. 87-88.

9. Чупринка Н. В. Проектування сумок з використанням автоматизованих систем / Н. В. Чупринка, А. О. Терстуйн // Тези доповідей XII Всеукраїнської наукової конференції молодих вчених та студентів, 25-26 квітня „Наукові розробки молоді на сучасному етапі”. – Т1, - К.:КНУТД, 2013. – С. 142-143.
10. Чупринка Н. В. Автоматизоване проектування деталей шкіргалантерейних виробів / Н. В. Чупринка // Тези доповідей XII Всеукраїнської наукової конференції молодих вчених та студентів, 25-26 квітня „Наукові розробки молоді на сучасному етапі”. – Т2 - К.: КНУТД, 2013. – С. 83-84.
11. Чупринка Н. В. Класифікація зовнішніх контурів деталей жіночих сумок для автоматизованого проектування цих деталей з використанням стандартних елементів / Н. В. Чупринка, С. С. Гаркавенко // Тези доповідей Всеукраїнської науково-практичної конференції «Проблеми легкої і текстильної промисловості України», 22-24 жовтня 2014, м. Херсон, – С. 63-65.
12. Чупринка Н. В. Автоматизоване проектування декоративних елементів для жіночих сумок. / Н. В. Чупринка, С. С. Гаркавенко // Тези доповідей Всеукраїнської науково-практичної конференції «Проблеми легкої і текстильної промисловості України», 22-24 жовтня 2014, м. Херсон, – С. 127-129.
13. Чупринка Н. В. Програмне забезпечення для генерування декоративних елементів жіночих сумок / Н.В. Чупринка // Тези доповідей XIII Всеукраїнської наукової конференції молодих вчених та студентів, 24-25 квітня 2014 р., „Наукові розробки молоді на сучасному етапі”. – Т1 - К.: КНУТД, 2014. – С. 85.
14. Чупринка Н. В. Програмне забезпечення для генерування декоративних елементів жіночих сумок / Н.В. Чупринка, С.С. Гаркавенко // Тези доповідей Всеукраїнської науково-практичної конференції студентів і молодих

- вчених, 15 квітня 2015 р., „Молодь-науці і виробництву –2015”. – Херсон. :ХНТУ, 2015. – С. 53-55.
15. Чупринка Н. В. Програмне забезпечення для генерування декоративних елементів на деталях жіночих сумок / Н. В. Чупринка // Тези доповідей XIV Всеукраїнської наукової конференції молодих вчених та студентів, 23-24 квітня 2015 р. „Наукові розробки молоді на сучасному етапі”. – Т1 - К.:КНУТД, 2015. – С. 72.
 16. Савіцький А. В. Стандартизація і сертифікація як важливий елемент управління якістю товарів та послуг / А. В. Савіцький // Вісник ХНУ. (Серія «Економічний простір»). – 2013. – № 41. – С. 239-247.
 17. Сумки, валізи, портфелі, ранці, папки, вироби дрібної шкіргалантереї. Загальні технічні умови : ДСТУ ГОСТ 28631:2006. – [Чинний від 2007-10-01]. – К. : Держспожив стандарт України, 2005. – 17 с. – (Національний стандарт України)
 18. Виробництво шкіргалантерейної продукції. Терміни та визначення : ДСТУ 3533-97. – [Чинний від 1998-07-01]. – К. : Держспожив стандарт України, 1996. – 15 с. – (Національний стандарт України)
 19. Вироби шкіргалантерейні. Терміни та визначення. ДСТУ 2198-93. – [Чинний від 1994-07-01]. – К. : Держспожив стандарт України, 1992. – 61 с. – (Національний стандарт України)
 20. Березовський В.С., Потієнко, В.О. та Завадський, І.О., 2009. Основи комп'ютерної графіки. Київ: ВНУ.
 21. Ванін В.В., Перевертун, В.В. та Надкернична, Т.М., 2006. Комп'ютерна інженерна графіка в середовищі AutoCAD. Київ: Каравела.
 22. Веселовська Г.В. та Ходакова, В.Є., 2015. Комп'ютерна графіка. Київ: Кондор.

- 23.Горобець С.М., 2006. Основи комп'ютерної графіки. Київ: Центр навчальної літератури. Manual of Shoemaking. /Under edition of R.G. Miller.Produced By the Trading Department Clarks, 1989. - 337 p.
- 24.Lectra-Marktführer in Brasilien//Schuh-Techn. Int.1995.-89, № 1 - 2. P.24
- 25.Gerber Hiera mit Partner//Schuh-Techn. Int. Schuh-Techn. + ABC. 1997.- 91, № 1 -2.-P. 6
- 26.Каменець С.Є., Васютинська В.В. Використання сучасних методів візуального дизайну та просторового моделювання для створення шкіргалантерейних виробів// Вісник Хмельницького національного університету. – 2020. - № 4. с. 199-205.
- 27.Столярова В. В., Каменець С. Є., Борщевська Н. М. Просторове моделювання та проектування аксесуарів і фурнітури виробів індустрії моди / Технології та дизайн № 3(36) 2020р. [Електронний ресурс] - Режим доступу:<https://drive.google.com/file/d/1DEkBLCjWLSRG23OkK0hgnKNDO-uZTZ6i/view?usp=sharing>.
- 28.Automation Testing Using Cucumber Tool and Selenium [Електронний ресурс] – Режим доступу: <http://www.softwaretestinghelp.com/cucumber-bdd-tool-selenium-tutorial-30/>
- 29.Helles, The Cucumber Book: Behaviour-Driven Development for Testers and Developers / M. Wynne, A. Hellesmy, The Pragmatic Bookshelf. - Jan. 2012. - 313 pp. - ISBN 978-1934356807
- 30.Corrca de Oliveira J., Costa Gouveia C., Quidute Filho R. “A way of Improving Test Automation CostEffectiveness.”, C.E.S.A.R., 2006, pp.1-5
- 31.Fewster M. “Common Mistakes in Test Automation.”, Grove Consultants, 2001, pp.1-7.
- 32.Marik B. “When Should a Test Be Automated?”, Testing Foundations, 1998, pp.1-20.

33. Rossmiller M. “6 Tips to Getting Started with Automated Testing.”, SmartBesar Software, 2011, pp.1- 7. 12. Vihari. S. “A System of Humanizing Test Automation Outlay Efficiency.” International Journal of Emerging Science and Engineering, May 2013, pp.74-78.
34. Autodesk ArtCAM Software End-of-Sale & End-of-Development FAQs. — URL: [https:// knowledge.autodesk.com/ search-result/ caas/ sfdcarticles/ sfdcarticles/ Autodesk-ArtCAM-Software-End-of-Sale-End-of-Development-FAQs.html](https://knowledge.autodesk.com/search-result/caas/sfdcarticles/sfdcarticles/Autodesk-ArtCAM-Software-End-of-Sale-End-of-Development-FAQs.html).
35. Software for shoe designers. // Arsutoria School— URL: [https:// www.arsutoriaschool.com/ software-for-shoe-designers-and-technical-drawing/](https://www.arsutoriaschool.com/software-for-shoe-designers-and-technical-drawing/)
36. Ishwor Khadka Software piracy: A study of causes, effects and preventive measures. Helsinki Metropolia University of Applied Sciences, 2015. Xanthopoulos S., Xinogalos S. A comparative analysis of cross-platform development approaches for mobile applications // Proceedings of the 6th Balkan Conference in Informatics. - 2013. - C. 213-220.
37. Lei K., Ma Y., Tan Z. Performance comparison and evaluation of web development technologies in php, python, and node. js // 2014 IEEE 17th international conference on computational science and engineering. - IEEE, 2014. - C. 661-668.
38. Tornincasa S. et al. The future and the evolution of CAD // Proceedings of the 14th international research/ expert conference: trends in the development of machinery and associated technology. - 2010. - T. 1. - №. 1. - C. 11-18.
39. Wang P. et al. Generating precise dependencies for large software // 2013 4th International Workshop on Managing Technical Debt (MTD). - IEEE, 2013. -C. 47-50.
40. Celerier J. M. A cross-platform development toolchain for jit-compilation in multimedia software // Proceedings of the 17th Linux Audio Conference. - 2019.

41. Kula R. G. et al. Do developers update their library dependencies? // Empirical Software Engineering. - 2018. - T. 23. - №. 1. - C. 384-417.
42. Bruun H. P. L. et al. PLM system support for modular product development // Computers in Industry. - 2015. - T. 67. - C. 97-111.
43. Meier U. et al. Twenty Years of PLM-the Good, the Bad and the Ugly // IFIP International Conference on Product Lifecycle Management. - Springer, Cham, 2017. - C. 69-77.
44. Stark J. Product lifecycle management (volume 2): the devil is in the details. - Springer, 2015.
45. Rani D., Ranjan R. K. A comparative study of SaaS, PaaS and IaaS in cloud computing // International Journal of Advanced Research in Computer Science and Software Engineering. - 2014. - T. 4. - №. 6.
46. Goyal S. Public vs private vs hybrid vs community-cloud computing: a critical review // International Journal of Computer Network and Information Security. - 2014. - T. 6. - №. 3. - C. 20.
47. Belov G., Scheithauer G., Mukhacheva E.A. On dimensional heuristics adapted for two-dimensional rectangular strip packing. Technical report. Dresden University. 2006. URL <http://www.math.tu-dresden.de/capad/>. Preprint MATH-NM-02-2006.
48. Belov G., Scheithauer G. A cutting plane algorithm for the one-dimensional cutting stock problem with multiple stock lengths // European Journal of Operational Research. 2002. 141. P. 274-294.
49. Berman P. Approximating rectilinear polygon cover problems / Berman P., Dasgupta B. // Algorithmica. #17(4). 1997. P. 331-356.
50. Blazewicz J., Hawryluk P., Walkowiak R. Using a tabu search approach for solving the two-dimensional irregular cutting problem // Annals of OR. 1993. 41(4). P. 313-325.

51. Bortfeldt A., Gehring H. Applying tabu search to container loading problems // Operations Research Proceedings. 1997, Springer, Berlin, 1998, P. 533538.
52. Boschetti M.A. The Two-Dimensional Finite Bin Packing Problem // Quarterly Journal of the Belgian, French and Italian Operations Research Societies, 2002.,
53. Bronnimann H. almost optimal set covers in finite VC-dimension. / Bronnimann H., Goodrich M. // Discrete comput. Geom., #14. 1995. P.263-279.
54. Burke E., Kendall G. Applying Ant Algorithms and the No Fit Polygon to the Nesting Problem // Accepted for the 1999 International Conference on Artificial Intelligence, Monte Carlo resort. Las Vegas. Nevada. USA. 1999.
55. Liu, X., Liu, J. M., Cao, A. X., and Yao, Z. L. (2015). HAPE3D-a new constructive algorithm for the 3D irregular packing problem. *Front. Inf. Technol. Electron. Eng.* 16 (5), 380–390. doi:10.1631/FITEE.1400421.
<https://link.springer.com/article/10.1631/FITEE.1400421>
56. Cherri, L. H., Cherri, A. C., and Soler, E. M. (2018). Mixed integer quadratically-constrained programming model to solve the irregular strip packing problem with continuous rotations. *J. Glob. Optim.* 72 (1), 89–107. doi:10.1007/s10898-018-0638-x. <https://link.springer.com/article/10.1007/s10898-018-0638-x>
57. Peralta, J., Andretta, M., and Oliveira, J. F. (2018). Solving irregular strip packing problems with free rotations using separation lines. *Pesqui. Oper.* 38 (2), 195–214. doi:10.1590/0101-7438.2018.038.02.0195.
<https://www.scielo.br/j/pope/a/RcXzqWKwBnL7QhcgkgNyZPv/?lang=en>
58. Stoyan, Y., Pankratov, A., and Romanova, T. (2016). Cutting and packing problems for irregular objects with continuous rotations: Mathematical modelling and non-linear optimization. *J. Operational Res. Soc.* 67 (5), 786–800. doi:10.1057/jors.2015.94.
<https://www.tandfonline.com/doi/abs/10.1057/jors.2015.94?journalCode=tjor20>
59. Wang, A., Hanselman, C. L., and Gounaris, C. E. (2018). A customized branch-and-bound approach for irregular shape nesting. *J. Glob. Optim.* 71 (4), 935–955.

doi:10.1007/s10898-018-0637-y.

<https://link.springer.com/article/10.1007/s10898-018-0637-y>

60. Guo, B., Ji, Y., Hu, J., Wu, F., and Peng, Q. (2019). Efficient free-form contour packing based on code matching strategy. *IEEE Access* 7, 57917–57926. doi:10.1109/ACCESS.2019.2914248.

<https://ieeexplore.ieee.org/abstract/document/8704207>

61. Романюк О.Н., 2001. Комп'ютерна графіка. Вінниця: Вінницький державний технічний університет.

62. Кондаков А.І. САПР технологічних процесів і виробництв. АСАДЕМА, 2007

63. Чупринка Н. В. Програмне забезпечення для генерування декоративних елементів у вигляді фігурних отворів на деталях жіночих сумок / Н.В. Чупринка // Тези доповідей XIV Всеукраїнської наукової конференції молодих вчених та студентів, 23 - 24 квітня 2015 р., Наукові розробки молоді на сучасному етапі” – Т2 - К.: КНУТД, 2015. – С. 52.

ДОДАТОК

**Лістинг програмного продукту автоматизованого проєктування
декоративних елементів для виробів шкіргалантереї**

```
nit UnKonetsky;  
  
interface  
  
uses  
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, ExtCtrls, ComCtrls, Printers;  
  
type  
TForm1 = class(TForm)  
PageControl1: TPageControl;  
TabSheet2: TTabSheet;  
Image1: TImage;  
ScrollBar1: TScrollBar;  
Image2: TImage;  
Image3: TImage;  
Image4: TImage;  
Image5: TImage;  
Image6: TImage;  
Image7: TImage;  
Image8: TImage;  
Image9: TImage;  
Image10: TImage;  
Image11: TImage;  
Image12: TImage;  
Image13: TImage;  
Image14: TImage;  
Image15: TImage;  
Image16: TImage;  
GroupBox1: TGroupBox;  
Button1: TButton;  
Label1: TLabel;  
Label2: TLabel;  
Label3: TLabel;  
Label4: TLabel;  
Edit1: TEdit;  
Edit2: TEdit;  
Edit3: TEdit;  
Edit4: TEdit;  
Label5: TLabel;
```

Edit5: TEdit;
Label6: TLabel;
Edit6: TEdit;
Image17: TImage;
Image18: TImage;
SaveDialog1: TSaveDialog;
Button20: TButton;
PrintDialog1: TPrintDialog;
Button21: TButton;
TabSheet1: TTabSheet;
Image19: TImage;

procedure Image2MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image3MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image4MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image5MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image6MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image7MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image8MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image9MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image10MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image11MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image12MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image13MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image14MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image15MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image16MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Button1Click(Sender: TObject);
procedure Image17MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image18MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Image12Click(Sender: TObject);
procedure Button19Click(Sender: TObject);

```

procedure Button20Click(Sender: TObject);
procedure Button21Click(Sender: TObject);

```

```

procedure FormCreate(Sender: TObject);
    private
    { Private declarations }
    public
    { Public declarations }
    end;
    var
    Form1: TForm1;
    implementation
    {$R *.dfm}
    Var N,Nd,p,KilksDet,NumDet,KilksPrikr,NumPrikr:integer;
    KilksPointDet,KilksPointPrikr:array[1..40]of integer;
    NameDet, NamePrikr:array[1..40]of string[20];
    Model:string[20];
    Xd,Yd:array[0..4000]of real;
    Xdet,Ydet:array[1..40,0..300]of real;
    XPrikr,YPrikr:array[1..40,0..600]of real;
    Xv,Yv,Xz,Yz:array[0..40]of real;
    Rv,Av,Rz,Az,A,B,L1,L2,Rz1,Rz2,A1,Fi,TetaV,TetaZ,t,mxy,R1,R2,DL_Pr,DL_Tr1,DL_Tr2,
    H_Pr,H_Tr:real;
    fl1,fl2,fl3,fl4,fl5,fl6,fl7,fl8,fl9,fl10,fl11,fl12,fl13,fl14,fl15,fl16,
    fl17,fl18,fl1_1,fl1_2,fl1_3,fl1_4,fl1_5,fl1_6,fl1_7,fl1_8,fl1_9,fl1_10,fl1_11,fl1_12,fl1_13,
    fl1_14,fl1_15,fl1_16,fl1_17,fl1_18:boolean;
    DLt,SHt,DLp,SHp,DLp1,SHp1,DLp2,SHp2,DL1,DL2,SH,R,A0,B0:integer;

```

```

    Procedure Det1;
    Var i:integer;
    Xce,Yce:real;
    begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    //Rv:=5; Rz:=40; N:=14;A:=5;
    Fi:=2*Pi/N;
    Av:=2*Rv*sin(Fi/2);
    If A>Av then A:=Av;
    Az:=2*Rz*sin(Fi/2);
    TetaV:=A*Fi/Av;
    TetaZ:=A*Fi/Az;
    For i:=0 to N-1 do
    begin
    Xz[2*i]:=Rz*cos(i*Fi-TetaZ/2);
    Xz[2*i+1]:=Rz*cos(i*Fi+TetaZ/2);
    Yz[2*i]:=Rz*sin(i*Fi-TetaZ/2);
    Yz[2*i+1]:=Rz*sin(i*Fi+TetaZ/2);

```

```

    Xv[2*i]:=Rv*cos(i*Fi-TetaV/2);
    Xv[2*i+1]:=Rv*cos(i*Fi+TetaV/2);
    Yv[2*i]:=Rv*sin(i*Fi-TetaV/2);
    Yv[2*i+1]:=Rv*sin(i*Fi+TetaV/2);
    end;
    For i:=0 to N-1 do
    begin
        Xd[4*i]:=Xv[2*i];
        Xd[4*i+1]:=Xz[2*i];
        Xd[4*i+2]:=Xz[2*i+1];
        Xd[4*i+3]:=Xv[2*i+1];
        Yd[4*i]:=Yv[2*i];
        Yd[4*i+1]:=Yz[2*i];
        Yd[4*i+2]:=Yz[2*i+1];
        Yd[4*i+3]:=Yv[2*i+1];
    end;
    Xd[4*N]:=Xd[0];
    Yd[4*N]:=Yd[0];
    Nd:=4*N+1;
    Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
    GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
    end;

```

```

procedure TForm1.Image2MouseDown(Sender: TObject; Button: TMouseButton;
    Shift: TShiftState; X, Y: Integer);
begin
    groupBox1.Visible:=True;
    Label5.Visible:=False;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    Label3.Caption:='Внутрішній радіус';
    Label4.Caption:='Ширину сторони';
    edit2.Text := '45';
    edit1.Text := '10';
    edit3.Text := '10';
    edit4.Text := '7';
    image1.Picture:=nil;
    Fl1:=True;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;

```

```

F110:=False;
F111:=False;
F112:=False;
F113:=False;
F114:=False;
F115:=False;
F116:=False;
F117:=False;
F118:=False;
end;

```

```

procedure TForm1.Image3MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := '9';
edit2.Text := '45';
edit3.Text := '10';
edit4.Text := '35';
image1.Picture:=nil;
groupBox1.Visible:=True;
Label4.Caption:='Пивіць B';
Label5.Visible:=False;
Edit5.Visible:=False;
Label6.Visible:=False;
Edit6.Visible:=False;
Label3.Caption:='Пивіць A';
Label4.Caption:='Пивіць B';
F12:=True;
F11:=False;
F13:=False;
F14:=False;
F15:=False;
F16:=False;
F17:=False;
F18:=False;
F19:=False;
F110:=False;
F111:=False;
F112:=False;
F113:=False;
F114:=False;
F115:=False;
F116:=False;
F117:=False;
F118:=False;
end;

```

```

procedure TForm1.Image4MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
  begin
    groupBox1.Visible:=True;
    Label3.Caption:='Пиввиць А';
    Label4.Caption:='Пиввиць В';
    Label5.Visible:=False;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    edit1.Text :='8';
    edit2.Text :='45';
    edit3.Text :='10';
    edit4.Text :='25';
    Fl3:=True;
    Fl1:=False;
    Fl2:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
  end;


```

```

procedure Det4;
  Var i,Ne,j,k,p,q:integer;
  Xce,Yce,H,Alfa,Fi1,Av1, Az1,TetaV,TetaZ,Rv1:real;
  Xc,Yc,Xe,Ye:array[0..40]of real;
  begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    //Rz:=40; N:=6; A:=15;
    N:=N*2; Ne:=19;
    B:=30; Alfa:=Pi/(Ne-1);
    If B>=Rz then B:=Rz-1;


```

```

    Fi:=2*Pi/N;
    H:=Rz-B;
    Av:=H*sin(Fi/2)/cos(Fi/2);
    If A>Av then A:=Av;
    Rv:=Av/sin(Fi/2);
    For i:=0 to N do
        begin
            Xv[i]:=Rv*cos(i*Fi-Fi/2);
            Yv[i]:=Rv*sin(i*Fi-Fi/2);
        end;
    // GrIm1(N+1,Xv,Yv,Xce,Yce);
    For i:=0 to N-1 do
        begin
            Xc[i]:=(Xv[i]+Xv[i+1])/2;
            Yc[i]:=(Yv[i]+Yv[i+1])/2;
        end;
    // GrIm1(N,Xc,Yc,Xce,Yce);
    For i:=0 to Ne-1 do
        begin
            Xe[i]:=B*cos(i*Alfa-Pi/2);
            Ye[i]:=A*sin(i*Alfa-Pi/2);
        end;
    // A1:=5; Rz1:=20;
    p:=1;
    Fil:=Fi/p;

    Rv1:=Rv;
    Av1:=2*Rv1*sin(Fil/2);
    If A1>Av1 then A1:=Av1;
    Az1:=2*Rz*sin(Fil/2);
    TetaV:=A1*Fi/Av1;
    TetaZ:=A1*Fi/Az1;
    q:=-1;
    For j:=0 to N-1 do
        begin

            if j mod 2=0 then
                For i:=0 to Ne-1 do
                    begin
                        q:=q+1;
                        Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
                        Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
                    end;

            if j mod 2=1 then
                begin
                    Xz[0]:=Rz1*cos(j*Fi-TetaZ/2);
                    Xz[1]:=Rz1*cos(j*Fi+TetaZ/2);
                    Yz[0]:=Rz1*sin(j*Fi-TetaZ/2);

```



```

        Yz[1]:=Rz1*sin(j*Fi+TetaZ/2);
        Xv[0]:=Rv1*cos(j*Fi-TetaV/2);
        Xv[1]:=Rv1*cos(j*Fi+TetaV/2);
        Yv[0]:=Rv1*sin(j*Fi-TetaV/2);
        Yv[1]:=Rv1*sin(j*Fi+TetaV/2);
        q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
        q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
        q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
        q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
        end;
        end;
        Nd:=q+2;
        Xd[Nd-1]:=Xd[0];
        Yd[Nd-1]:=Yd[0];
        Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
        GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
        end;

procedure TForm1.Image5MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
    groupBox1.Visible:=True;
    Label3.Caption:='Піввісь A';
    Label4.Caption:='Піввісь B';
    Label5.Visible:=False;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    edit1.Text :='8';
    edit2.Text :='45';
    edit3.Text :='10';
    edit4.Text :='25';
    Fl4:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;

```

```

Fl17:=False;
Fl18:=False;
end;

```

```

procedure Det5;
  Var i,Ne,j,k,p,q:integer;
  Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,Rz1,Rv1:real;
  Xc,Yc,Xe,Ye:array[0..40]of real;
  begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    Rz:=40; N:=4;
    N:=N*2; A:=15; Ne:=19;
    B:=30; Alfa:=Pi/(Ne-1);
    If B>=Rz then B:=Rz-1;
    Fi:=2*Pi/N;
    H:=Rz-B;
    Av:=H*sin(Fi/2)/cos(Fi/2);
    If A>Av then A:=Av;
    Rv:=Av/sin(Fi/2);
    For i:=0 to N do
      begin
        Xv[i]:=Rv*cos(i*Fi-Fi/2);
        Yv[i]:=Rv*sin(i*Fi-Fi/2);
      end;
    // GrIm1(N+1,Xv,Yv,Xce,Yce);
    For i:=0 to N-1 do
      begin
        Xc[i]:=(Xv[i]+Xv[i+1])/2;
        Yc[i]:=(Yv[i]+Yv[i+1])/2;
      end;
    // GrIm1(N,Xc,Yc,Xce,Yce);
    For i:=0 to Ne-1 do
      begin
        Xe[i]:=B*cos(i*Alfa-Pi/2);
        Ye[i]:=A*sin(i*Alfa-Pi/2);
      end;
    A1:=5; p:=1;
    Fi1:=Fi/p;
    Rz1:=30;
    Rv1:=Rv;
    Av1:=2*Rv1*sin(Fi1/2);
    If A1>Av1 then A1:=Av1;
    Az1:=2*Rz1*sin(Fi1/2);
    TetaV:=A1*Fi/Av1;
    TetaZ:=A1*Fi/Az1;
    q:=-1;

```

```

For j:=0 to N-1 do
  begin
    if j mod 2=0 then
      For i:=0 to Ne-1 do
        begin
          q:=q+1;
          Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
          Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
        end;

        if j mod 2=1 then
          begin
            Xz[0]:=Rz1*cos(j*Fi-TetaZ/2);
            Xz[1]:=Rz1*cos(j*Fi+TetaZ/2);
            Yz[0]:=Rz1*sin(j*Fi-TetaZ/2);
            Yz[1]:=Rz1*sin(j*Fi+TetaZ/2);
            Xv[0]:=Rv1*cos(j*Fi-TetaV/2);
            Xv[1]:=Rv1*cos(j*Fi+TetaV/2);
            Yv[0]:=Rv1*sin(j*Fi-TetaV/2);
            Yv[1]:=Rv1*sin(j*Fi+TetaV/2);
            q:=q+1; Xd[q]:=Xv[0];      Yd[q]:=Yv[0];
            q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2;  Yd[q]:=(Yz[0]+Yz[1])/2;
            q:=q+1; Xd[q]:=Xv[1];      Yd[q]:=Yv[1];
          end;
        end;
        Nd:=q+2;
        Xd[Nd-1]:=Xd[0];
        Yd[Nd-1]:=Yd[0];
      Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
      GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
    end;
  end;

```

```

procedure TForm1.Image6MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
  begin
    groupBox1.Visible:=True;
    Label3.Caption:='Пиввіць А';
    Label4.Caption:='Пиввіць В';
    Label5.Visible:=False;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    edit1.Text :='8';
    edit2.Text :='45';
    edit3.Text :='10';
    edit4.Text :='25';
    Fl5:=True;
  end;

```

```

    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    end;

    procedure Det6;
    Var i,Ne,j,k,q,m:integer;
    Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,Rv1:real;
    Xc,Yc,Xe,Ye:array[0..40]of real;
    begin
        Xce:=form1.Image1.Width/2;
        Yce:=form1.Image1.Height/2;
        //N:=7; Rz:=40; A:=15; B:=30;
        N:=N*2; Ne:=19;
        Alfa:=Pi/(Ne-1);
        If B>=Rz then B:=Rz-1;
        Fi:=2*Pi/N;
        H:=Rz-B;
        Av:=H*sin(Fi/2)/cos(Fi/2);
        If A>Av then A:=Av;
        Rv:=Av/sin(Fi/2);
        For i:=0 to N do
            begin
                Xv[i]:=Rv*cos(i*Fi-Fi/2);
                Yv[i]:=Rv*sin(i*Fi-Fi/2);
            end;
        // GrIm1(N+1,Xv,Yv,Xce,Yce);
        For i:=0 to N-1 do
            begin
                Xc[i]:=(Xv[i]+Xv[i+1])/2;
                Yc[i]:=(Yv[i]+Yv[i+1])/2;
            end;
        // GrIm1(N,Xc,Yc,Xce,Yce);
        For i:=0 to Ne-1 do
            begin
                Xe[i]:=B*cos(i*Alfa-Pi/2);
                Ye[i]:=A*sin(i*Alfa-Pi/2);
            end;
    end;

```

```

end;

    Fi1:=Fi/p;
    //p:=2; Rz1:=20; A1:=12;
    Rv1:=Rv;
    Av1:=2*Rv1*sin(Fi1/2);
    A1:=Av1;
    Az1:=2*Rz1*sin(Fi1/2);
    TetaV:=A1*Fi1/Av1;
    TetaZ:=A1*Fi1/Az1;
    q:=-1;
    For j:=0 to N-1 do
        begin

            if j mod 2=0 then
                For i:=0 to Ne-1 do
                    begin
                        q:=q+1;
                        Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
                        Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
                    end;

                    if j mod 2=1 then
                        for m:=0 to p-1 do
                            begin
                                Xz[0]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
                                Xz[1]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
                                Yz[0]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
                                Yz[1]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
                                Xv[0]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m));
                                Xv[1]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m+1));
                                Yv[0]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m));
                                Yv[1]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m+1));
                                q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
                                q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
                                q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
                                q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
                            end;
                        end;
                    Nd:=q+2;
                    Xd[Nd-1]:=Xd[0];
                    Yd[Nd-1]:=Yd[0];
                Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
                GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
            end;
        end;
    end;

procedure TForm1.Image7MouseDown(Sender: TObject; Button: TMouseButton;
    Shift: TShiftState; X, Y: Integer);
    begin

```

```

    groupBox1.Visible:=True;
    Label3.Caption:='Піввісь A';
    Label4.Caption:='Піввісь B';
    Label5.Caption:='Зовнішній радіус l';
    Label5.Visible:=True;
    Edit5.Visible:=True;
    Label6.Visible:=True;
    Edit6.Visible:=True;
    image1.Picture:=nil;
    edit1.Text :='7';
    edit2.Text :='40';
    edit3.Text :='15';
    edit4.Text :='30';
    edit5.Text :='35';
    edit6.Text :='3';
    Fl6:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
    end;

```

```

    procedure Det7;
    Var i,Ne,j,k,q,m:integer;
    Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,Rv1:real;
    Xc,Yc,Xe,Ye:array[0..40]of real;
    begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    // Rz:=40; N:=6; A:=15; B:=30;
    N:=N*2; Ne:=19;
    Alfa:=Pi/(Ne-1);
    If B>=Rz then B:=Rz-1;
    Fi:=2*Pi/N;
    H:=Rz-B;

```

```

Av:=H*sin(Fi/2)/cos(Fi/2);
If A>Av then A:=Av;
Rv:=Av/sin(Fi/2);
For i:=0 to N do
begin
Xv[i]:=Rv*cos(i*Fi-Fi/2);
Yv[i]:=Rv*sin(i*Fi-Fi/2);
end;
// GrIm1(N+1,Xv,Yv,Xce,Yce);
For i:=0 to N-1 do
begin
Xc[i]:=(Xv[i]+Xv[i+1])/2;
Yc[i]:=(Yv[i]+Yv[i+1])/2;
end;
// GrIm1(N,Xc,Yc,Xce,Yce);
For i:=0 to Ne-1 do
begin
Xe[i]:=B*cos(i*Alfa-Pi/2);
Ye[i]:=A*sin(i*Alfa-Pi/2);
end;
//A1:=15; p:=3;
Fi1:=Fi/p;
// Rz1:=25;
Rv1:=Rv;
Av1:=2*Rv1*sin(Fi1/2);
A1:=Av1;
Az1:=2*Rz1*sin(Fi1/2);
TetaV:=A1*Fi1/Av1;
TetaZ:=A1*Fi1/Az1;
q:=-1;
For j:=0 to N-1 do
begin

if j mod 2=0 then
For i:=0 to Ne-1 do
begin
q:=q+1;
Xd[q]:=Xe[i]*cos(j*Fi)-Ye[i]*sin(j*Fi)+Xc[j];
Yd[q]:=Xe[i]*sin(j*Fi)+Ye[i]*cos(j*Fi)+Yc[j];
end;

if j mod 2=1 then
for m:=0 to p-1 do
begin
Xz[0]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
Xz[1]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
Yz[0]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+1));
Yz[1]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ/2+TetaZ*(m+2));
Xv[0]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m));

```

```

Xv[1]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m+1));
Yv[0]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m));
Yv[1]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m+1));
q:=q+1; Xd[q]:=Xv[0];      Yd[q]:=Yv[0];
q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2;  Yd[q]:=(Yz[0]+Yz[1])/2;
q:=q+1; Xd[q]:=Xv[1];      Yd[q]:=Yv[1];
      end;
      end;
      Nd:=q+2;
      Xd[Nd-1]:=Xd[0];
      Yd[Nd-1]:=Yd[0];

```

```

Scala( Nd, Xd, Yd, Form1.Image1.Width-40, Form1.Image1.Height-40, mxy);
GrIm1(Nd, Xd, Yd, Xce, Yce, mxy);
      end;

```

```

procedure TForm1.Image8MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
  groupBox1.Visible:=True;
  Label3.Caption:='Піввісь A';
  Label4.Caption:='Піввісь B';
  Label5.Caption:='Зовнішній радіус l';
  Label4.Visible:=True;
  Edit4.Visible:=True;
  Label5.Visible:=True;
  Edit5.Visible:=True;
  Label6.Visible:=True;
  Edit6.Visible:=True;
  image1.Picture:=nil;
  edit1.Text :='7';
  edit2.Text :='40';
  edit3.Text :='15';
  edit4.Text :='30';
  edit5.Text :='35';
  edit6.Text :='3';
  Fl7:=True;
  Fl1:=False;
  Fl2:=False;
  Fl3:=False;
  Fl4:=False;
  Fl5:=False;
  Fl6:=False;
  Fl8:=False;
  Fl9:=False;
  Fl10:=False;
  Fl11:=False;
  Fl12:=False;

```



```

    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
    end;

    procedure Det8;
    Var i,q:integer;
    Xce,Yce:real;
    begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    // Rv:=5; Rz:=40; N:=14;
    // A:=5;
    Fi:=2*Pi/N;
    Av:=2*Rv*sin(Fi/2);
    //If A>Av then
    A:=Av;
    Az:=2*Rz*sin(Fi/2);
    TetaV:=A*Fi/Av;
    TetaZ:=A*Fi/Az;
    q:=-1;
    For i:=0 to N-1 do
    begin
    Xz[0]:=Rz*cos(i*Fi-TetaZ/2);
    Xz[1]:=Rz*cos(i*Fi+TetaZ/2);
    Yz[0]:=Rz*sin(i*Fi-TetaZ/2);
    Yz[1]:=Rz*sin(i*Fi+TetaZ/2);
    Xv[0]:=Rv*cos(i*Fi-TetaV/2);
    Xv[1]:=Rv*cos(i*Fi+TetaV/2);
    Yv[0]:=Rv*sin(i*Fi-TetaV/2);
    Yv[1]:=Rv*sin(i*Fi+TetaV/2);

    q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
    q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2; Yd[q]:=(Yz[0]+Yz[1])/2;
    q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
    end;
    q:=q+1;
    Xd[q]:=Xd[0];
    Yd[q]:=Yd[0];
    Nd:=q+1;
    Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
    GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
    end;

```

```
procedure TForm1.Image9MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
```

```
begin
  groupBox1.Visible:=True;
  Label5.Visible:=False;
  Edit4.Visible:=False;
  Label4.Visible:=False;
  Edit5.Visible:=False;
  Label6.Visible:=False;
  Edit6.Visible:=False;
  image1.Picture:=nil;
  Label3.Caption:='Внутрішній радіус';
  edit2.Text := '45';
  edit1.Text := '10';
  edit3.Text := '10';
  edit4.Text := '7';
  Fl8:=True;
  Fl1:=False;
  Fl2:=False;
  Fl3:=False;
  Fl4:=False;
  Fl5:=False;
  Fl6:=False;
  Fl7:=False;
  Fl9:=False;
  Fl10:=False;
  Fl11:=False;
  Fl12:=False;
  Fl13:=False;
  Fl14:=False;
  Fl15:=False;
  Fl16:=False;
  Fl17:=False;
  Fl18:=False;
end;
```

```
procedure Det9;
  Var i,q:integer;
  Xce,Yce,TetaZ1,TetaZ2,Az1,Az2:real;
begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rv:=5; Rz1:=40; Rz2:=20; N:=6;
  N:=N*2;
  Fi:=2*Pi/N;
  Av:=2*Rv*sin(Fi/2);
```

```

        A:=Av;
        Az1:=2*Rz1*sin(Fi/2);
        Az2:=2*Rz2*sin(Fi/2);
        TetaV:=A*Fi/Av;
        TetaZ1:=A*Fi/Az1;
        TetaZ2:=A*Fi/Az2;
        q:=-1;
        For i:=0 to N-1 do
            begin
                if i mod 2=0 then
                    begin
                        TetaZ:=TetaZ1;
                        Rz:=Rz1;
                    end
                else
                    begin
                        TetaZ:=TetaZ2;
                        Rz:=Rz2;
                    end ;
                Xz[0]:=Rz*cos(i*Fi-TetaZ/2);
                Xz[1]:=Rz*cos(i*Fi+TetaZ/2);
                Yz[0]:=Rz*sin(i*Fi-TetaZ/2);
                Yz[1]:=Rz*sin(i*Fi+TetaZ/2);
                Xv[0]:=Rv*cos(i*Fi-TetaV/2);
                Xv[1]:=Rv*cos(i*Fi+TetaV/2);
                Yv[0]:=Rv*sin(i*Fi-TetaV/2);
                Yv[1]:=Rv*sin(i*Fi+TetaV/2);

                q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
                q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2; Yd[q]:=(Yz[0]+Yz[1])/2;
                q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
            end;
            q:=q+1;
            Xd[q]:=Xd[0];
            Yd[q]:=Yd[0];
            Nd:=q+1;
        Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
        GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
    end;

```

```

procedure TForm1.Image10MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
    groupBox1.Visible:=True;
    Label5.Visible:=False;
    Edit4.Visible:=True;
    Label4.Visible:=True;

```

```

    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    Label2.Caption:='Зовнішній радіус 1';
    Label3.Caption:='Зовнішній радіус 2';
    Label4.Caption:='Внутрішній радіус';
    edit2.Text := '45';
    edit1.Text := '10';
    edit3.Text := '30';
    edit4.Text := '17';
    Fl9:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
    end;

```

```

    procedure Det10;
    Var i,q:integer;
    Xce,Yce,TetaZ1,TetaZ2,Az1,Az2:real;
    begin
        Xce:=Form1.Image1.Width/2;
        Yce:=Form1.Image1.Height/2;
        // Rv:=5; Rz1:=20; Rz2:=40; N:=6;
        N:=N*2;
        Fi:=2*Pi/N;
        Av:=2*Rv*sin(Fi/2);
        A:=Av;
        Az1:=2*Rz1*sin(Fi/2);
        Az2:=2*Rz2*sin(Fi/2);
        TetaV:=A*Fi/Av;
        TetaZ1:=A*Fi/Az1;
    end;

```

```

TetaZ2:=A*Fi/Az2;
q:=-1;
For i:=0 to N-1 do
begin
if i mod 2=0 then
begin
TetaZ:=TetaZ1;
Rz:=Rz1;
end
else
begin
TetaZ:=TetaZ2;
Rz:=Rz2;
end ;
Xz[0]:=Rz*cos(i*Fi-TetaZ/2);
Xz[1]:=Rz*cos(i*Fi+TetaZ/2);
Yz[0]:=Rz*sin(i*Fi-TetaZ/2);
Yz[1]:=Rz*sin(i*Fi+TetaZ/2);
Xv[0]:=Rv*cos(i*Fi-TetaV/2);
Xv[1]:=Rv*cos(i*Fi+TetaV/2);
Yv[0]:=Rv*sin(i*Fi-TetaV/2);
Yv[1]:=Rv*sin(i*Fi+TetaV/2);
q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
if i mod 2=0 then
begin
q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2; Yd[q]:=(Yz[0]+Yz[1])/2;
end
else
begin
q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
end;
q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
end;
q:=q+1;
Xd[q]:=Xd[0];
Yd[q]:=Yd[0];
Nd:=q+1;
Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
end;

procedure TForm1.Image1MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
groupBox1.Visible:=True;
Label5.Visible:=False;
Edit4.Visible:=True;
Label4.Visible:=True;

```

```

    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    Label2.Caption:='Зовнішній радіус 1';
    Label3.Caption:='Зовнішній радіус 2';
    Label4.Caption:='Внутрішній радіус';
    edit2.Text := '45';
    edit1.Text := '6';
    edit3.Text := '30';
    edit4.Text := '17';
    Fl10:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl15:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
    end;

```

```

    procedure Det11;
    Var i,Ne,j,k,q,m,Nd:integer;
    Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,TetaZ1,Rz1,Rv1:real;
    Xc,Yc,Xe,Ye:array[0..40]of real;
    begin
    Xce:=Form1.Image1.Width/2;
    Yce:=Form1.Image1.Height/2;
    Rz:=40; N:=6; A:=15; B:=30;
    N:=N*2; Alfa:=Pi/(Ne-1);
    // If B>=Rz then B:=Rz-1;
    Fi:=2*Pi/N;
    H:=Rv;
    Av:=H*sin(Fi/2)/cos(Fi/2);
    A:=Av; Av1:=Av;
    Az:=2*Rz*sin(Fi/2);
    TetaZ:=A*Fi/Az;
    Rv:=Av/sin(Fi/2);

```

```

    A1:=A; Az1:=Az;
    //p:=3;
    Fi1:=Fi/p;
    Rz1:=Rz;
    Rv1:=Rv;

    TetaV:=A1*Fi1/Av1;
    TetaZ1:=A1*Fi1/Az1;
    q:=-1;
    For j:=0 to N-1 do
        begin

            if j mod 2=1 then
                for m:=0 to p-1 do
                    begin
                        Xz[0]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+1));
                        Xz[1]:=Rz1*cos(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+2));
                        Yz[0]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+1));
                        Yz[1]:=Rz1*sin(j*Fi+Fi1*m-Fi/2-p*TetaZ1/2+TetaZ1*(m+2));
                        Xv[0]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m));
                        Xv[1]:=Rv1*cos(j*Fi-p*TetaV/2+TetaV*(m+1));
                        Yv[0]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m));
                        Yv[1]:=Rv1*sin(j*Fi-p*TetaV/2+TetaV*(m+1));
                        q:=q+1; Xd[q]:=Xv[0];      Yd[q]:=Yv[0];
                        q:=q+1; Xd[q]:=(Xz[0]+Xz[1])/2;  Yd[q]:=(Yz[0]+Yz[1])/2;
                        q:=q+1; Xd[q]:=Xv[1];      Yd[q]:=Yv[1];
                    end;
                end;
            Nd:=q+2;
            Xd[Nd-1]:=Xd[0];
            Yd[Nd-1]:=Yd[0];
        Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
        GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
        end;

procedure TForm1.Image12MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);

    begin
        groupBox1.Visible:=True;
        Label5.Visible:=False;
        Edit4.Visible:=True;
        Label4.Visible:=True;
        Edit5.Visible:=False;
        Label6.Visible:=False;
        Edit6.Visible:=False;
        Label2.Caption:='Зовнішній радіус ';
        Label3.Caption:='Внутрішній радіус';
        Label4.Caption:='Кількість ланцюгів ';

```

```

edit2.Text := '45';
edit1.Text := '6';
edit3.Text := '20';
edit4.Text := '3';
  Fl1:=True;
  Fl1:=False;
  Fl2:=False;
  Fl3:=False;
  Fl4:=False;
  Fl5:=False;
  Fl6:=False;
  Fl7:=False;
  Fl8:=False;
  Fl9:=False;
  Fl10:=False;
  Fl12:=False;
  Fl13:=False;
  Fl14:=False;
  Fl15:=False;
  Fl16:=False;
  Fl17:=False;
  Fl18:=False;
  end;

  procedure Det12;
  Var i,Ne,j,k,q,m,Nd:integer;
  Xce,Yce,H,Alfa,Fi1,Av1,A1, Az1,TetaV,TetaZ,TetaZ1,Rz1,Rv1,p:real;
  Xc,Yc,Xe,Ye:array[0..40]of real;
  begin
  Xce:=Form1.Image1.Width/2;
  Yce:=Form1.Image1.Height/2;
  //Rz:=40; N:=6;
  N:=N*2;
  // A:=15; Ne:=19; B:=30;
  // Alfa:=Pi/(Ne-1);
  //If B>=Rz then B:=Rz-1;
  Fi:=2*Pi/N;
  H:=Rv;
  Av:=2*Rv*sin(Fi/2);
  A:=Av;
  Az:=2*Rz*sin(Fi/2);
  TetaZ:=A*Fi/Az;
  // Rv:=Av/sin(Fi/2);
  // A1:=15;
  //t:=1/4;
  Fi1:=Fi/t;
  // Rz1:=25;

  Rz1:=Rz; Rv1:=Rv;

```



```

Av1:=2*Rv1*sin(Fi1/2);
//If A1>Av1 then
  A1:=Av1;
  // A:=A1;
Az1:=2*Rz1*sin(Fi1/2);
TetaV:=A*Fi/Av1;
TetaZ1:=A*Fi/Az;
  q:=-1;
  For j:=0 to N-1 do
    begin

      if j mod 2=0 then
        begin
          Xz[0]:=Rz*cos(j*Fi-TetaZ/2);
          Xz[1]:=Rz*cos(j*Fi+TetaZ/2);
          Yz[0]:=Rz*sin(j*Fi-TetaZ/2);
          Yz[1]:=Rz*sin(j*Fi+TetaZ/2);
          Xv[0]:=Rv*cos(j*Fi-TetaV/2);
          Xv[1]:=Rv*cos(j*Fi+TetaV/2);
          Yv[0]:=Rv*sin(j*Fi-TetaV/2);
          Yv[1]:=Rv*sin(j*Fi+TetaV/2);
          q:=q+1; Xd[q]:=Xv[0]; Yd[q]:=Yv[0];
          q:=q+1; Xd[q]:=Xz[0]; Yd[q]:=Yz[0];
          q:=q+1; Xd[q]:=Xz[1]; Yd[q]:=Yz[1];
          q:=q+1; Xd[q]:=Xv[1]; Yd[q]:=Yv[1];
        end;
      end;
      Nd:=q+2;
      Xd[Nd-1]:=Xd[0];
      Yd[Nd-1]:=Yd[0];
      Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
      GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
    end;

```

```

procedure TForm1.Image13MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);

```

```

  begin
    groupBox1.Visible:=True;
    Label4.Visible:=True;
    Edit4.Visible:=True;
    Label5.Visible:=False;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    Label2.Caption:='Зовнішній радіус ';

```

```

Label3.Caption:='Внутрішній радіус';
Label4.Caption:='Параметр  $t$  (0,25 - 2)';
edit2.Text := '45';
edit1.Text := '6';
edit3.Text := '20';
edit4.Text := '1';
Fl12:=True;
Fl1:=False;
Fl2:=False;
Fl3:=False;
Fl4:=False;
Fl5:=False;
Fl6:=False;
Fl7:=False;
Fl8:=False;
Fl9:=False;
Fl10:=False;
Fl11:=False;
Fl13:=False;
Fl14:=False;
Fl15:=False;
Fl16:=False;
Fl17:=False;
Fl18:=False;
end;

```

```

procedure TForm1.Image14MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
groupBox1.Visible:=True;
Label5.Visible:=True;
Edit4.Visible:=True;
Label4.Visible:=True;
Edit5.Visible:=True;
Label6.Visible:=False;
Edit6.Visible:=False;
image1.Picture:=nil;
Label2.Caption:='Зовнішній радіус 1';
Label3.Caption:='Зовнішній радіус 2';
Label4.Caption:='Внутрішній радіус';
Label5.Caption:='Кількість ланцюгів';

```

```
edit1.Text := '6';  
edit2.Text := '45';  
edit3.Text := '30';  
edit4.Text := '17';  
edit5.Text := '3';  
F113:=True;  
F11:=False;  
F12:=False;  
F13:=False;  
F14:=False;  
F15:=False;  
F16:=False;  
F17:=False;  
F18:=False;  
F19:=False;  
F110:=False;  
F111:=False;  
F112:=False;  
F114:=False;  
F115:=False;  
F116:=False;  
F117:=False;  
F118:=False;  
end;
```

```
procedure Det14;  
Var A,Fi,Xce,Yce:real;  
i,Nd:integer;  
begin  
// Nd:=9; R:=40;  
Xce:=Form1.Image1.Width/2;
```

```

Yce:=Form1.Image1.Height/2;
    Fi:=2*Pi/N;
    N:=N+1;
    For i:=0 to N-1 do
        begin
            Xd[i]:=Rz*cos(i*Fi);
            Yd[i]:=Rz*sin(i*Fi);
        end;
    Scala( N, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
    GrIm1(N,Xd,Yd,Xce,Yce,mxy);
    end;

procedure TForm1.Image15MouseDown(Sender: TObject; Button: TMouseButton;
    Shift: TShiftState; X, Y: Integer);
    begin
        groupBox1.Visible:=True;
        Label3.Visible:=False;
        Edit3.Visible:=False;
        Label5.Visible:=False;
        Edit4.Visible:=False;
        Label4.Visible:=False;
        Edit5.Visible:=False;
        Label6.Visible:=False;
        Edit6.Visible:=False;
        image1.Picture:=nil;
        Label2.Caption:='Зовнішній радіус ';
        edit1.Text :='6';
        edit2.Text :='45';
        Fl14:=True;
        Fl1:=False;
        Fl2:=False;
        Fl3:=False;
        Fl4:=False;
    end;

```

```

Fl5:=False;
Fl6:=False;
Fl7:=False;
Fl8:=False;
Fl9:=False;
Fl10:=False;
Fl11:=False;
Fl12:=False;
Fl13:=False;
Fl15:=False;
Fl16:=False;
Fl17:=False;
Fl18:=False;
    end;

```

```

Procedure NewPoint(Xa,Ya,Xb,Yb:real; K:integer; Var Xc,Yc:real);

```

```

    Var d,d1:real;

```

```

        begin

```

```

            d:=sqrt(sqr(Xa-Xb)+ sqr(Ya-Yb));

```

```

                d1:=d/k;

```

```

                    Xc:=d1*(Xb-Xa)/d+Xa;

```

```

                    Yc:=d1*(Yb-Ya)/d+Ya;

```

```

                end;

```

```

        procedure Det15;

```

```

            Var A,Fi,Xce,Yce,Xc,Yc:real;

```

```

                i,k,q:integer;

```

```

            Ydz,Xdz,Ydv,Xdv:array[0..50]of real;

```

```

                begin

```

```

                    // N:=9; Rz:=40; Rv:=5; k:=25;

```

```

                    Xce:=Form1.Image1.Width/2;

```

```

                    Yce:=Form1.Image1.Height/2;

```

```

                    Fi:=2*Pi/N;

```

```

        N:=N+1;
        For i:=0 to N-1 do
            begin
                Xdz[i]:=Rz*cos(i*Fi);
                Ydz[i]:=Rz*sin(i*Fi);
            end;
        For i:=0 to N-1 do
            begin
                Xdv[i]:=Rv*cos(i*Fi);
                Ydv[i]:=Rv*sin(i*Fi);
            end;
        q:=-1;
        For i:=0 to N-2 do
            begin
                q:=q+1; Xd[q]:=Xdv[i]; Yd[q]:=Ydv[i];
                q:=q+1; NewPoint(Xdz[i],Ydz[i],Xdz[i+1],Ydz[i+1], p,Xd[q],Yd[q]);
                q:=q+1; NewPoint(Xdz[i+1],Ydz[i+1],Xdz[i],Ydz[i], p,Xd[q],Yd[q]);
            end;
        q:=q+1; Xd[q]:=Xd[0]; Yd[q]:=Yd[0];
        Nd:=q+1;
        Scala( Nd, Xd,Yd,Form1.Image1.Width-40,Form1.Image1.Height-40,mxy);
        GrIm1(Nd,Xd,Yd,Xce,Yce,mxy);
        end;

procedure TForm1.Image1MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
    groupBox1.Visible:=True;
    Label3.Visible:=True;;
    Edit3.Visible:=True;;
    Label5.Visible:=False;
    Edit4.Visible:=True;;
    Label4.Visible:=True;;

```

```
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    Label2.Caption:='Зовнішній радіус ';
    Label4.Caption:='Коеф. розхилу(>2) ';
    edit1.Text :='6';
    edit2.Text :='45';
    edit3.Text :='26';
    edit4.Text :='4';
    Fl15:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;
    Fl14:=False;
    Fl16:=False;
    Fl17:=False;
    Fl18:=False;
    end;
```

procedure TForm1.Image17MouseDown(Sender: TObject; Button: TMouseButton;

```

Shift: TShiftState; X, Y: Integer);
    begin
    groupBox1.Visible:=True;
    Label3.Visible:=True;;
    Edit3.Visible:=True;;
    Label5.Visible:=False;
    Edit4.Visible:=True;;
    Label4.Visible:=True;;
    Edit5.Visible:=False;
    Label6.Visible:=False;
    Edit6.Visible:=False;
    image1.Picture:=nil;
    Label2.Caption:='Зовнішній радіус ';
    Label3.Caption:='Внутрішній радіус ';
    Label4.Caption:='Коеф. розхилу(>2) ';
    edit1.Text :='9';
    edit2.Text :='45';
    edit3.Text :='16';
    edit4.Text :='24';
    Fl16:=True;
    Fl1:=False;
    Fl2:=False;
    Fl3:=False;
    Fl4:=False;
    Fl5:=False;
    Fl6:=False;
    Fl7:=False;
    Fl8:=False;
    Fl9:=False;
    Fl10:=False;
    Fl11:=False;
    Fl12:=False;
    Fl13:=False;

```



```

F114:=False;
F115:=False;
F117:=False;
F118:=False;
end;

```

```

procedure TForm1.Image18MouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
groupBox1.Visible:=True;
Label3.Visible:=True;;
Edit3.Visible:=True;;
Label5.Visible:=False;
Edit4.Visible:=True;;
Label4.Visible:=True;;
Edit5.Visible:=False;
Label6.Visible:=False;
Edit6.Visible:=False;
image1.Picture:=nil;
Label1.Caption:='Сторона A';
Label2.Caption:='Сторона B ';
Label3.Caption:='Верхній розхил L1 ';
Label4.Caption:='Нижній розхил L2(L2>=L1) ';
edit1.Text :='20';
edit2.Text :='45';
edit3.Text :='6';
edit4.Text :='24';
F117:=True;
F11:=False;
F12:=False;
F13:=False;
F14:=False;

```

```
Fl5:=False;  
Fl6:=False;  
Fl7:=False;  
Fl8:=False;  
Fl9:=False;  
Fl10:=False;  
Fl11:=False;  
Fl12:=False;  
Fl13:=False;  
Fl14:=False;  
Fl15:=False;  
Fl16:=False;  
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin  
    Button20.Visible:=True;  
    Button21.Visible:=True;  
    GroupBox1.Visible:=False;  
    If Fl1 then  
        begin  
            N:=StrToInt(edit1.Text);  
            Rz:=StrToInt(edit2.Text);  
            Rv:=StrToInt(edit3.Text);  
            A:=StrToInt(edit4.Text);  
            Det1;  
        end  
    else If Fl2 then  
        begin  
            N:=StrToInt(edit1.Text);  
            Rz:=StrToInt(edit2.Text);  
            A:=StrToInt(edit3.Text);  
            B:=StrToInt(edit4.Text);
```

```
        Det2;
        end
    else If Fl3 then
        begin
            N:=StrToInt(edit1.Text);
            Rz:=StrToInt(edit2.Text);
            A:=StrToInt(edit3.Text);
            B:=StrToInt(edit4.Text);
            Det3;
            end
    else If Fl4 then
        begin
            N:=StrToInt(edit1.Text);
            Rz:=StrToInt(edit2.Text);
            A:=StrToInt(edit3.Text);
            Rz1:=StrToInt(edit4.Text);
            A1:=StrToInt(edit5.Text);
            Det4;
            end
    else If Fl5 then
        begin
            N:=StrToInt(edit1.Text);
            Rz:=StrToInt(edit2.Text);
            A:=StrToInt(edit3.Text);
            Rz1:=StrToInt(edit4.Text);
            A1:=StrToInt(edit5.Text);
            Det5;
            end
    else If Fl6 then
        begin
            N:=StrToInt(edit1.Text);
            Rz:=StrToInt(edit2.Text);
            A:=StrToInt(edit3.Text);
```

```

    B:=StrToInt(edit4.Text);
Rz1:=StrToInt(edit5.Text);
    p:=StrToInt(edit6.Text);
        Det6;
        end
    else If Fl7 then
        begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
A:=StrToInt(edit3.Text);
B:=StrToInt(edit4.Text);
Rz1:=StrToInt(edit5.Text);
p:=StrToInt(edit6.Text);
        Det7;
        end
    else If Fl8 then
        begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rv:=StrToInt(edit3.Text);
        Det8;
        end
    else If Fl9 then
        begin
N:=StrToInt(edit1.Text);
Rz1:=StrToInt(edit2.Text);
Rz2:=StrToInt(edit3.Text);
Rv:=StrToInt(edit4.Text);
        Det9;
        end
    else If Fl10 then
        begin
N:=StrToInt(edit1.Text);

```

```

Rz1:=StrToInt(edit2.Text);
Rz2:=StrToInt(edit3.Text);
Rv:=StrToInt(edit4.Text);
    Det10;
    end
else If Fl11 then
    begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rv:=StrToInt(edit3.Text);
p:=StrToInt(edit4.Text);
        Det11;
        end
    else If Fl12 then
        begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rv:=StrToInt(edit3.Text);
t:=StrToFloat(edit4.Text);
            Det12;
            end
        else If Fl13 then
            begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rz1:=StrToInt(edit3.Text);
Rv:=StrToInt(edit4.Text);
p:=StrToInt(edit5.Text);
                Det13;
                end
            else If Fl14 then
                begin
N:=StrToInt(edit1.Text);

```

```
Rz:=StrToInt(edit2.Text);
    Det14;
    end
else If Fl15 then
    begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rv:=StrToInt(edit3.Text);
p:=StrToInt(edit4.Text);
    Det15;
    end
else If Fl16 then
    begin
N:=StrToInt(edit1.Text);
Rz:=StrToInt(edit2.Text);
Rv:=StrToInt(edit3.Text);
p:=StrToInt(edit4.Text);
    Det16;
    end
else If Fl17 then
    begin
A:=StrToInt(edit1.Text);
B:=StrToInt(edit2.Text);
L1:=StrToInt(edit3.Text);
L2:=StrToInt(edit4.Text);
    Det17;
    end
end;
```

,